Problems in Signal Processing and Inference on Graphs

Problems in Signal Processing and Inference on Graphs

A dissertation presented
by

Ameya Agaskar

to
The School of Engineering and Applied Sciences
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy
in the subject of

Engineering Sciences

Harvard University
Cambridge, Massachusetts
May 2015

Dissertation Advisor
Professor Yue M. Lu

Author
Ameya Agaskar

Problems in Signal Processing and Inference on Graphs

Abstract

Modern datasets are often massive due to the sharp decrease in the cost of collecting and storing data. Many are endowed with relational structure modeled by a graph, an object comprising a set of points and a set of pairwise connections between them. A "signal on a graph" has elements related to each other through a graph—it could model, for example, measurements from a sensor network. In this dissertation we study several problems in signal processing and inference on graphs.

We begin by introducing an analogue to Heisenberg's time-frequency uncertainty principle for signals on graphs. We use spectral graph theory and the standard extension of Fourier analysis to graphs. Our spectral graph uncertainty principle makes precise the notion that a highly localized signal on a graph must have a broad spectrum, and vice versa.

Next, we consider the problem of detecting a random walk on a graph from noisy observations. We characterize the performance of the optimal detector through the (type-II) error exponent, borrowing techniques from statistical physics to develop a lower bound exhibiting a phase transition. Strong performance is only guaranteed when the signal to noise ratio exceeds twice the random walk's entropy rate. Monte Carlo simulations show that the lower bound is quite close to the true exponent.

Next, we introduce a technique for inferring the source of an epidemic from observations at a few nodes. We develop a Monte Carlo technique to simulate the infection process, and use statistics computed from these simulations to approximate the likelihood, which we then maximize to locate the source.

We further introduce a logistic autoregressive model (ALARM), a simple model for binary processes on graphs that can still capture a variety of behavior. We demonstrate its simplicity by showing how to easily infer the underlying graph structure from measurements; a technique versatile enough that it can work under model mismatch.

Finally, we introduce the exact formula for the error of the randomized Kaczmarz algorithm, a linear system solver for sparse systems, which often arise in graph theory. This is important because, as we show, existing performance bounds are quite loose.

Contents

Citations to Previously Published Work

Most of Chapter 2 was published as the following paper:

> A. Agaskar and Y. M. Lu. A spectral graph uncertainty principle. *IEEE Transactions on Information Theory*, 59(7):4338–4356, July 2013.

Chapter 3 has been prepared for publication as:

> A. Agaskar and Y. M. Lu. Optimal Detection of Random Walks on Graphs: Performance Analysis via Statistical Physics. Submitted to *IEEE Transactions on Information Theory*, 2015.

Part of Chapter 3 was published as:

> A. Agaskar and Y. M. Lu. Detecting random walks hidden in noise: Phase transition on large graphs. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc. (ICASSP)*, 2013.

Most of Chapter 4 was published as the following two papers:

> A. Agaskar and Y. M. Lu. A fast Monte Carlo algorithm for source localization on graphs. In *SPIE Wavelets*, 2013.
>
> A. Agaskar and Y. M. Lu. ALARM: A logistic auto-regressive model for binary processes on networks. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2013.

Most of the material in Chapter 5 was published as the following two papers:

> A. Agaskar, C. Wang, and Y. M. Lu. Randomized Kaczmarz algorithms: Exact MSE analysis and optimal sampling probabilities. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2014. *Student Paper Award*.
>
> C. Wang, A. Agaskar, and Y. M. Lu. Randomized Kaczmarz algorithm for inconsistent linear systems: An exact MSE analysis. In *Sampling Theory and Applications (SampTA)*, 2015.

Some additional material in Chapter 5 is being prepared for publication as:

> A. Agaskar, C. Wang, and Y. M. Lu. Randomized Kaczmarz Algorithm and its Cousins: Exact MSE Analysis and Asymptotic Dynamics. *Manuscript*, 2015.

Acknowledgments

I MUST START BY expressing gratitude to the many people who have played a part in my education: these include my mother, Rohini Kothare Agaskar, who patiently taught me everything from long division to trigonometric identities and basic derivatives long before I "got to them" in math class; my high school calculus and physics teachers, Miss Elizabeth Drews and Mrs. Mary Yeomans, respectively, who helped cultivate my love of math and physics and ensured that I was overprepared when I began my undergraduate studies at Cornell; Prof. Lang Tong, who supervised my M.Eng. project at Cornell and guided me through getting my first journal publication, Larry Horowitz and Dan Bliss (now Professor Bliss at Arizona State University) of the Advanced Sensor Techniques group at MIT Lincoln Laboratory for hours of technical discussions in their offices, and my Ph.D. advisor Prof. Yue M. Lu.

Yue, of course, deserves his own paragraph. His first semester as a professor was my first as a Ph.D. student. Since day one, he has been a patient and effective teacher. He has a special and strong kind of empathy that is so important in a teacher: he knows exactly when to push, and exactly when to cut slack, in order to maximize productivity and minimize stress. I have learned a great deal from him.

Dr. Gary Hatke of the Advanced Sensor Techniques group at MIT Lincoln Laboratory has been an extremely supportive group leader. I would also like to recognize my colleagues at Lincoln Lab for lots of interesting technical discussions and Ph.D. advice often drawn from personal experience: Larry and Dan, who I men-

tioned earlier, Keith Forsythe, Swaroop Appadwedula, Cathy Keller, Adam Margetts, Rachel Learned, Christ Richmond, Ben Miller, and Steve Smith. I am truly looking forward to "returning to the fold" and working full time at the Laboratory.

I am also grateful to my friends and colleagues at Harvard, including Behtash Babadi, Chenhui Hu, Stanley Chan, Ariana Minot, Hongyao Ma, Chuang Wang, who gave me spirited technical discussions, feedback on papers and presentations, lunchtime company, sugar-delivery mechanisms from all around the world, and more. I would also like to thank my sister, Manishika, for her support during and after her time in Cambridge.

I would like to thank Prof. Vahid Tarokh and Prof. Roger Brockett for serving on my thesis committee, and to Prof. Li Na for helpful discussions.

I am also grateful to Cindy Sullivan and Gina O'Donovan at Lincoln Lab and Gioia Sweetland at Harvard for administrative support.

I would also like to acknowledge MIT Lincoln Laboratory, whose Lincoln Scholars program funded my doctoral studies, the Department of the Air Force for funding the Laboratory and allowing it to invest in my education, and the American people, whose taxes are the ultimate source of these funds and whose consent grants the Government the power to use them in this manner.

Finally, more than mere thanks are owed to my dear wife, Dr.[1] Cheryl Sorace-Agaskar. She is brilliant and beautiful, as anyone who has met her will know. But

[1] She won the race.

nearly a decade by her side has taught me that her patience, tenacity, empathy, and creativity make her someone to emulate. And to my son, Simon, if you are reading this, I hope I've already told you just how much a single smile from you could brighten a tough day, back when you were a baby and Dad was a graduate student.

This work was sponsored by the Department of the Air Force under Contract FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

To my family.

Chapter 1

Introduction

MODERN datasets are often massive due to the sharp decrease in the cost of collecting and storing data. Many of these datasets possess some sort of relational structure, where a datum may be connected to another through proximity, affinity, or, more concretely, a direct communication link [18, 55, 79, 131]. A mathematical model for this sort of relational structure is a *graph*, which is an object consisting of a set of points and information about pairwise connections between the points [32]. In general, these connections might be directed or weighted. Several graphs are illustrated in Figure 1.1.

Graphs can model both neuronal and anatomical connectivity in the brain, human interaction networks through which diseases spread, sensor networks with distributed communication and processing, regulatory networks in biological systems, transportation networks through which traffic flows, and many other real-world

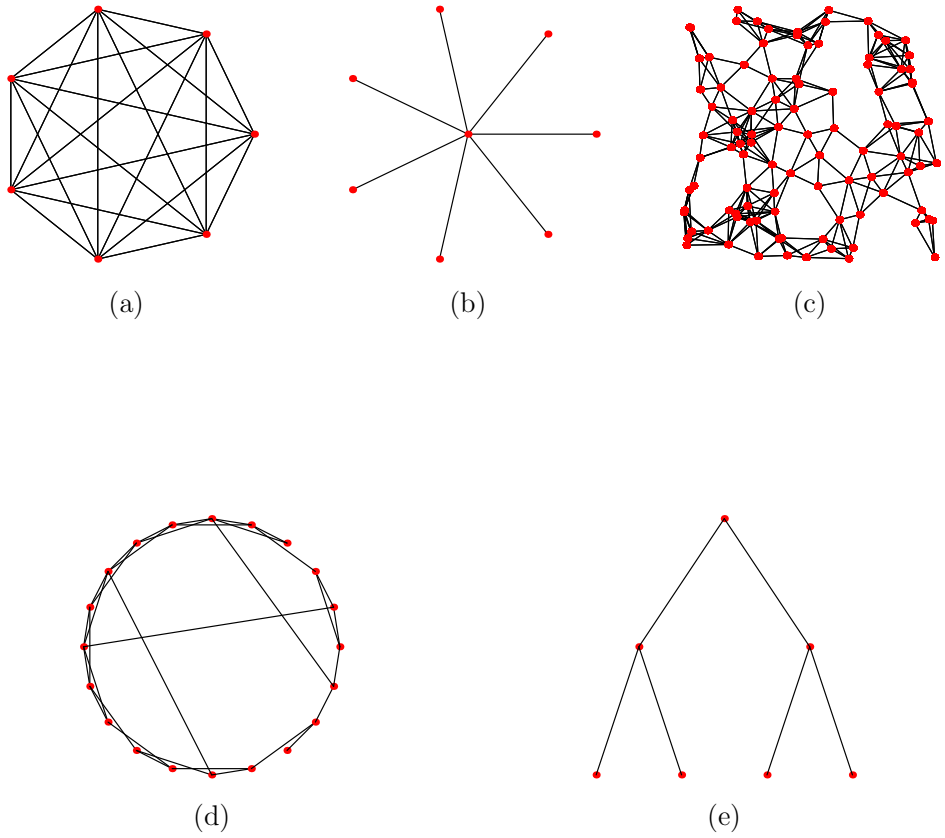


Figure 1.1: A graph represents relational structure between various nodes. Several types of graphs are illustrated here: (a) a complete graph, a degenerate graph in which all nodes are connected to each other; (b) a star graph, in which a central node is connected to all others; (c) a random geometric graph, in which nodes are selected as random points and connected to nearby nodes; (d) a small world graph, a model in which most connections are local but a few long-distance connections greatly reduce average distance; (e) a binary tree graph.

systems. More abstractly, graphs often have utility as natural discretizations of the complicated manifold structure underlying high dimensional signals. When the relational structure is paired with measurements on the nodes of the network, we have a *signal on a graph*.

Whatever the network, the processing we would like to perform on such a signal may be very similar to standard signal processing on 1-D time series or 2-D images, *e.g.* filtering and smoothing, detection (hypothesis testing), parameter estimation, sampling and reconstruction, or compression. On the other hand, we also may wish to answer a question unique to this setting: can we learn the network structure itself from measurements that are bound to it somehow? In any of these cases, there are unique challenges and opportunities due to the presence of the graph structure. Exploiting or inferring this structure is the common thread running through this dissertation.

1.1 Outline of dissertation

In this dissertation, we study several problems related to signal processing and inference on graphs, in which we either exploit the graph structure or try to it from our observations. The outline of the dissertation is as follows.

In recent years, there has been considerable effort to extend the concepts of classical signal processing to the graph setting. One common desire is localized,

multiscale transforms that extract components of graph signals that are appropriately localized on the graph itself and in the graph spectral domain, which is defined by the eigenvectors of the Laplacian matrix. In Chapter 2, we show that there is an ultimate limit for this localization. Analogous to the Heisenberg uncertainty principle, which prohibits a classical signal from being highly localized in both time and frequency, we find an uncertainty principle for signals on graphs that prevents signals from being highly localized on the graph and in the spectral domain. We define graph and spectral spreads that are analogous to the time and frequency spreads defined by Heisenberg, and show how to find the "uncertainty curve" that defines the smallest graph spread for any given spectral spread. We also reveal an intriguing connection to diffusion processes.

How much can knowledge of graph structure help us to detect very weak signals? This is the question we consider in Chapter 3, where we study the problem of detecting a random walk. Specifically, given a sequence of observations from every node in a graph, we seek to distinguish between two hypotheses: (a) every observation is just meaningless zero-mean Gaussian noise, or (b) an agent is undergoing a random walk on the graph and raises the mean of the observation at its location at each time. We characterize the performance of the optimal detector by the (type-II) error exponent: the decay rate of the miss probability with increasing observations under a false alarm constraint. We use a connection to statistical physics to develop a lower bound that can be computed using techniques borrowed

from that field. Our fully rigorous analysis uses large deviations theory to show that the lower bound exhibits a phase transition: strong performance is only guaranteed when the signal to noise ratio exceeds twice the entropy rate of the Markov chain. Monte Carlo simulations show that the lower bound is quite close to the true exponent.

There are many dynamical processes on graphs in which the graph structure controls the behavior of the process by requiring interactions to primarily occur between nodes that are connected to each other. In Chapter 4 we consider two such models. The first is the standard susceptible-infected (SI) model from epidemiology, which models the spread of an infection through a network. This is a very old model, but only recently has anyone considered the problem of detecting the source of an infection from later observations. We introduce our technique, which can perform this inference given a sequence of snapshots from only a small number of nodes by performing Monte Carlo simulations to estimate the likelihood maximizing it. We also introduce our own model for general binary dynamic processes on graphs: a logistic autoregressive model (ALARM). In this model, a node's value at time t is determined by a linear combination of its neighbors values at preceding times through a logistic link function. This can capture all sorts of behavior: nodes can positively or negatively influence their neighbors, and a node's value can be made to be "sticky". We illustrate some of this behavior, and show how the generality of the model allows us to perform inference when the true dynamics are

unknown.

Next, we consider a problem with broad applications: solving a linear system. In particular, in Chapter 5, we consider the performance of the randomized Kaczmarz algorithm, a linear system solver that is particularly suited for sparse matrices of that sort that often appear in graph problems (*e.g.* adjacency, Laplacian, and incidence matrices). The Kaczmarz algorithm is a well-known iterative algorithm that has been studied and used for decades. It can suffer from convergence difficulties under certain conditions, but recently a randomized version of the algorithm was proposed that has provably exponential convergence. We introduce the first exact formula for the mean squared error (MSE) of the algorithm at each time step; this formula works for both noiseless and noisy linear systems. We show how to compute the limiting "error floor" in the noisy case, and the error exponent measuring the decay rate of the MSE in the noiseless case. We consider the problem of optimizing the randomization of the algorithm for fastest convergence. We also show numerically that existing performance bounds are very weak, making our exact formula useful in practice.

Finally, in Chapter 6 we summarize the main contributions of this dissertation and discuss future research directions.

Chapter 2

A Spectral Graph Uncertainty Principle

HEISENBERG'S UNCERTAINTY principle is a cornerstone of signal processing. The simple inequality [53, 128],

$$\Delta_t^2 \Delta_\omega^2 \geq \frac{1}{4}, \tag{2.1}$$

in which Δ_t^2 and Δ_ω^2 measure the "time spread" and "frequency spread" of some signal, respectively, is one way to precisely characterize a general principle with far-reaching consequences: that a signal cannot be concentrated in both time and frequency. This is illustrated in Figure 2.1.

In this chapter, we establish analogous uncertainty principles for signals defined on graphs. The study of signals on graphs, and the extension of classical signal processing techniques to such nonstandard domains, has received growing interest

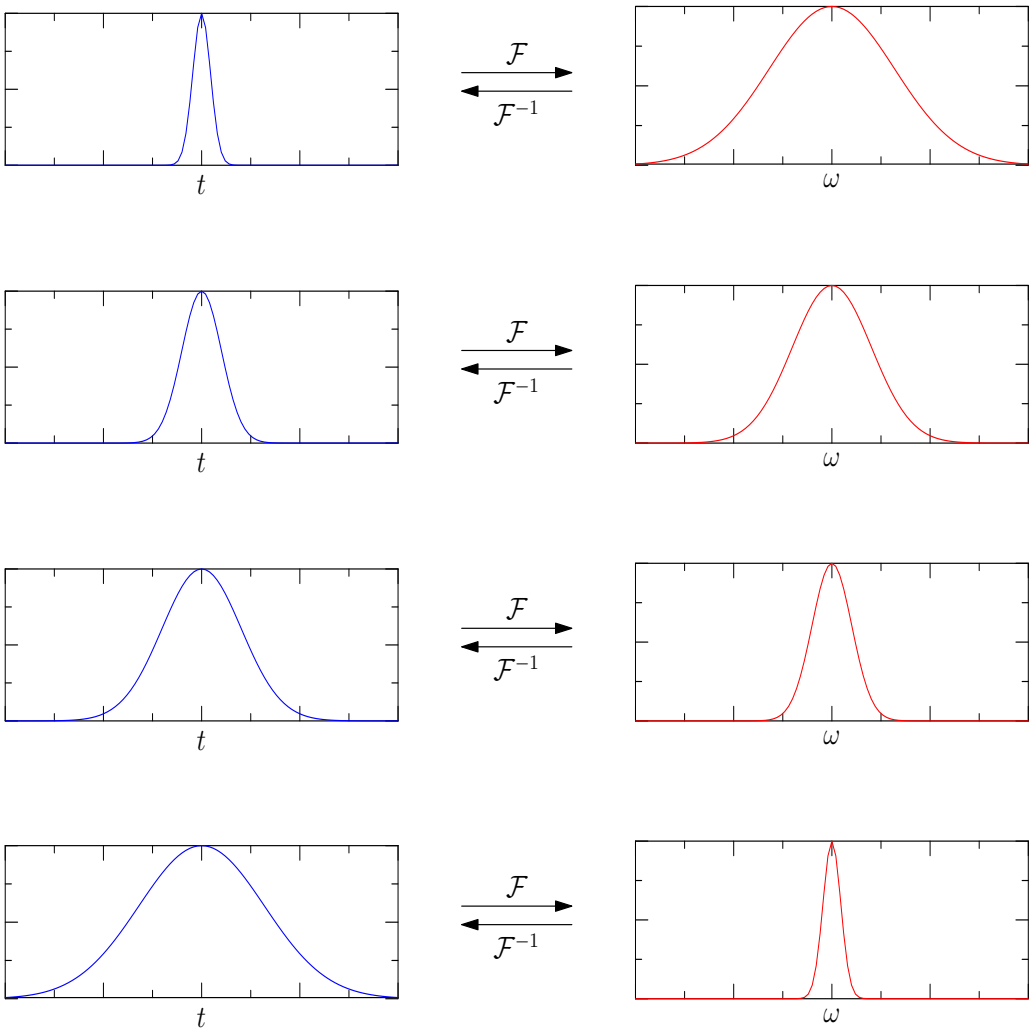


Figure 2.1: The Heisenberg-Weyl uncertainty principle causes the Fourier transform of a function to narrow as the function itself widens. Gaussian functions such as the one in this figure are the only ones that achieve the bound exactly.

in the past few years (see, *e.g.*, [23,29,33,34,43,62,76,92,102]). These studies are often motivated (and enabled) by the deluge of modern data collected on various technological, social, biological, and informational networks [79]. The efficient acquisition, representation, and analysis of such high-dimensional graph-based data present challenges that should be addressed by the development of new signal processing theories and tools. By using analogies between traditional and graph-based signal processing, researchers can bootstrap their intuitive understanding of standard signal processing problems to gain some insight into the more complicated graph-based problems. This is the basic motivation for developing an uncertainty principle for signals on graphs.

Related Work

Uncertainty principles date back to Heisenberg [53], who in 1927 proved a result that Weyl and Pauli soon afterward generalized to (2.1). It was also shown that the bound in (2.1) is achievable by Gaussian-shaped functions and frequency modulations thereof. A lifetime later, analogous results were found for discrete-time signals as well [25, 69]. Similar uncertainty principles have also been established on the unit sphere S^d [57] and, in more abstract settings, on compact Riemannian manifolds [49].

In a different line of work, Donoho and Stark [44] introduced a new concept of uncertainty related to signal support size. They showed that a length N

discrete-time signal with support set $\mathcal{T}$ in the time domain and support set $\mathcal{W}$ in the frequency domain satisfies $|\mathcal{T}|\,|\mathcal{W}| \geq N$. This bound is a *nonlocal* uncertainty principle—it limits the cardinality of a signal's time and frequency support sets, even if each is the disjoint union of far-flung subsets. Further studied in, *e.g*, [26, 27, 46], these nonlocal uncertainty principles laid the foundation for sparse signal recovery from partial measurements.

In the same vein of the classical (and local) uncertainty principle stated in (2.1), we have been studying the following question: given an arbitrary graph, to what extent can a signal be simultaneously localized on that graph and in the "frequency" domain? To obtain the spectral representation of these signals, we use the standard approach of treating the eigenvectors of the graph Laplacian operator [56] as a Fourier basis. The Laplacian encodes a notion of smoothness on a graph [18] and is analogous to the Laplace-Beltrami operator on a manifold [17].

The analogy between the spectral decomposition of graph Laplacians and the standard Fourier transform has been used to extend the concept of bandlimited sampling to signals defined on graphs [102] and in the construction of wavelet transforms on graphs [34, 62, 92]. In the latter case, as pointed out in [92], a desirable property of the wavelet transforms is that the dictionary elements (*i.e.*, wavelets) should be well-localized in the graph and spectral domains. Our results provide a way to precisely quantify this desideratum, as well as its fundamental limit.

Contributions

We begin in Section 2.1 with a review of some basic concepts in graph theory, including the definition of the graph Laplacian matrix and its spectral decomposition. After justifying the use of the Laplacian eigenvectors as a Fourier basis on graphs, we define in Section 2.1.3 the *graph spread* about a vertex u_0, $\Delta^2_{g,u_0}(\boldsymbol{x})$, and the *spectral spread*, $\Delta^2_s(\boldsymbol{x})$, of a signal $\boldsymbol{x}$ defined on a graph. These two quantities, which we first introduced in some preliminary work [2, 3], are defined in analogy to the standard time and frequency spreads, respectively.

In [2], we developed a lower bound on the product of Δ^2_{g,u_0} and Δ^2_s analogous to (2.1). However, the bound was not tight and applied only under restrictive conditions for the graph and the signal on it. In [3] we took a new approach to characterize a more general and precise relationship between the two kinds of uncertainty. In [7], we continued this line of investigation, and provided a rigorous basis for the arguments presented in [3], in addition to some new results. This chapter mainly follows the results of [7].

The main contributions are developed in Section 2.2, where we characterize the uncertainty bound, in Section 2.3, where we analyze the bound when applied to special families of graphs, and in Section 2.4, where we reveal a connection between diffusion processes and the uncertainty bound. The main results are summarized as follows:

1. *Convexity of the feasible region*: We prove that, when the underlying graph is

connected and contains at least three vertices, the feasibility region of all possible pairs $(\Delta_s^2(\boldsymbol{x}), \Delta_{g,u_0}^2(\boldsymbol{x}))$ is a *bounded* and *convex* set.

2. *Characterization of the uncertainty curve*: We provide a complete characterization of the curve

$$\gamma_{u_0}(s) = \min_{\boldsymbol{x}} \Delta_{g,u_0}^2(\boldsymbol{x}) \text{ subject to } \Delta_s^2(\boldsymbol{x}) = s,$$

which forms the lower boundary of the feasibility region. Studying $\gamma_{u_0}(s)$, which we will refer to as the *uncertainty curve*, is important because it is a fundamental bound analogous to the classical uncertainty bound (2.1). Theorem 2.1 states that each point on the uncertainty curve is achieved by an eigenvector associated with the smallest eigenvalue of a particular matrix-valued function $\boldsymbol{M}(\alpha)$. Varying the parameter α allows one to "trace" and obtain the entire curve $\gamma_{u_0}(s)$. A rigorous and complete proof of Theorem 2.1 is provided. Based the convexity of $\gamma_{u_0}(s)$, we show in Section 2.2.3 that the *sandwich algorithm* [111] can be used to efficiently produce a piecewise linear approximation for the uncertainty curve that differs from the true curve by at most ε (under a suitable error metric) and requires solving $\mathcal{O}(\varepsilon^{-1/2})$ typically sparse eigenvalue problems.

3. *Special graph families*: The uncertainty curves for several special families of graphs are investigated in Section 2.3. For complete graphs and star graphs, we derive closed-form formulas for the uncertainty curves $\gamma_{u_0}(s)$. For Erdős-Rényi random graphs [50, 51], we develop an analytical approximation for the expected value of $\gamma_{u_0}(s)$, which is shown through experiment to be very accurate.

4. *Diffusion process on a graph*: In Section 2.4, we reveal an intriguing connection between the classical uncertainty principle for functions on the real line and our results for signals on graphs. In the classical case, the solution to the heat equation $\frac{du}{dt} = \frac{d^2u}{dy^2}$ starting at $t = 0$ as an impulse is a Gaussian function with a variance that grows linearly with t; this solution achieves the Heisenberg uncertainty bound (2.1). We first show experimental results indicating that a diffusion process starting with an impulse on a graph follows the graph uncertainty curve very closely (though not, in general, exactly.) We then prove in Proposition 2.4 that the match is exact for the special cases of a star graph or a complete graph. We further prove in Proposition 2.5 that for general graphs, under a simple condition on the distance function on the graph, the first two derivatives of the uncertainty curve and the curve traced by the diffusion process match at the point corresponding to $t = 0$.

2.1 Mathematical Formulation

2.1.1 Graphs, Signals, and Notation

We define a simple, undirected graph as $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_N\}$ is a set of N vertices and $E = \{e_1, e_2, \ldots, e_M\}$ is the set of M edges. Each edge is an unordered pair of two different vertices $u, v \in V$, and we will use the notation $u \sim v$ to indicate that u and v are connected by an edge. The fundamental structure of a graph G can be captured by its *adjacency matrix* $\boldsymbol{A} = [a_{ij}]_{ij}$, where $a_{ij} = 1$

if there is an edge between v_i and v_j, and $a_{ij} = 0$ otherwise. As defined, the diagonal of $\boldsymbol{A}$ is always zero because a simple graph may contain no loops (*i.e.*, edges connecting one vertex to itself), and $\boldsymbol{A}$ is symmetric because the graph is undirected. (A common generalization is to consider a weighted graph, where each edge e_m is associated with a positive "weight" w_m. In this chapter we only consider unweighted graphs, but other researchers have recently explored extending our results to the weighted graph case [100].)

The degree of a vertex v, denoted by $\deg(v)$, is the number of edges incident upon that vertex. We define $\boldsymbol{D}$ as the diagonal matrix that has the vertex degrees on its diagonal, *i.e.*,

$$\boldsymbol{D} \stackrel{\text{def}}{=} \operatorname{diag}\left\{\deg(v_1), \deg(v_2), \ldots, \deg(v_N)\right\}. \tag{2.2}$$

To quantify the graph-domain spread of a signal, we will need a notion of distance, denoted by $d(u, v)$, between any pair of vertices u and v on the graph. A simple choice is to use the *geodesic distance* [56], in which case $d(u, v)$ is the length of the shortest path connecting the two vertices. In this work, we only consider connected graphs, so $d(u, v)$ is always finite. Other distance functions have been proposed in the literature, including the resistance distance [77] and the diffusion distance [34]. Our subsequent discussions are not confined to any particular choice of the distance function. The only requirement is that $d(u, v)$ should form a semi-metric: namely, $d(u, v) \geq 0$ with equality if and only if $u = v$, and $d(u, v) = d(v, u)$.

A finite-energy signal defined on the graph $\boldsymbol{x} \in \ell^2(G)$ is a mapping from the set

of vertices to $\mathbb{R}$. It can be treated as a vector in $\mathbb{R}^N$, and so any such signal will be denoted by a boldface variable. There is a natural inner product on $\ell^2(G)$ defined by $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{y}^T \boldsymbol{x}$, which induces a norm $\|\boldsymbol{x}\| = \sqrt{\boldsymbol{x}^T \boldsymbol{x}}$. We will denote the value of $\boldsymbol{x}$ at vertex v by $x(v)$. An impulse at $v \in V$, *i.e.*, a signal that has value 1 at v and 0 everywhere else, will be denoted as $\boldsymbol{\delta}_v$.

2.1.2 The Laplacian Matrix, Graph Fourier Transforms, and Signal Processing on Graphs

As mentioned earlier, the graph Laplacian matrix plays an important role in this work. There are several different definitions of the Laplacian matrix commonly used in the literature. The unnormalized Laplacian matrix [56] is given by $\boldsymbol{L}_{\text{unnorm}} \stackrel{\text{def}}{=} \boldsymbol{D} - \boldsymbol{A}$, where $\boldsymbol{D}$ and $\boldsymbol{A}$ are the degree matrix in (2.2) and the adjacency matrix, respectively. In this work, we find it more convenient to use the normalized Laplacian matrix [32], defined as

$$\begin{aligned} \boldsymbol{L}_{\text{norm}} &\stackrel{\text{def}}{=} \boldsymbol{D}^{-1/2} \boldsymbol{L}_{\text{unnorm}} \boldsymbol{D}^{-1/2} \\ &= \boldsymbol{I} - \boldsymbol{D}^{-1/2} \boldsymbol{A} \boldsymbol{D}^{-1/2}. \end{aligned}$$

The choice of unnormalized or normalized Laplacian makes no essential difference to our analysis in Section 2.2. The latter is chosen because it leads to simpler expressions in some of our derivations. For notational simplicity, we will drop the subscript in $\boldsymbol{L}_{\text{norm}}$, calling it $\boldsymbol{L}$ in what follows.

Intuitively, the Laplacian matrix is analogous to the continuous Laplacian operator $-\nabla^2$ or $-\frac{d^2}{dy^2}$ on the real line. In fact, when the underlying graph is a line or a cycle, $\boldsymbol{L}$ provides the standard stencil approximation for the second-order differentiation operator. The same holds for higher-dimensional lattices. In more general settings where the graphs are formed by sampling an underlying continuous manifold, the Laplacian matrix converges at high sampling densities to the Laplace-Beltrami operator, a differential geometric analogy to the second derivative [17].

By construction, $\boldsymbol{L}$ is a real symmetric matrix. We can therefore diagonalize $\boldsymbol{L}$ as

$$\boldsymbol{L} = \boldsymbol{F}\boldsymbol{\Lambda}\boldsymbol{F}^T, \tag{2.3}$$

where $\boldsymbol{F}$ is an orthogonal matrix whose columns are the eigenvectors of $\boldsymbol{L}$, and $\boldsymbol{\Lambda} \stackrel{\text{def}}{=} \operatorname{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ is a diagonal matrix of eigenvalues, which are all real. $\boldsymbol{L}$ can be shown to be positive semidefinite with rank less than N, so we can order the eigenvalues as $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_N$.

A large number of the topological properties of a graph can be inferred from the spectrum of its graph Laplacian [32]. For example, a graph is connected (meaning that a path can always be found connecting one vertex to the other) if and only if the smallest eigenvalue ($\lambda_1 = 0$) has multiplicity one. The corresponding unit-norm eigenvector $\boldsymbol{f}_1$ is defined by

$$f_1(v) = \sqrt{\frac{\deg(v)}{\sum_{u \in V} \deg(u)}}, \tag{2.4}$$

where $\deg(v)$ is the degree of the vertex v. The second-smallest eigenvalue, known

as the *algebraic connectivity* [56], is a measure of the degree of connectedness of the graph: higher values mean a more tightly-knit graph; the associated eigenvector is known as the *Fiedler vector* [127] and provides a sort of canonical partition of a graph through the signs of its entries. One can also show that the maximum possible eigenvalue of $\boldsymbol{L}$ is equal to 2, attained only by bipartite graphs. (These are graphs with two mutually exclusive subsets of vertices U_0 and U_1 such that every edge connects a vertex in U_0 to a vertex in U_1.)

Given a signal $\boldsymbol{x} \in \ell^2(G)$, we can represent it in terms of the eigenvectors of $\boldsymbol{L}$ by computing

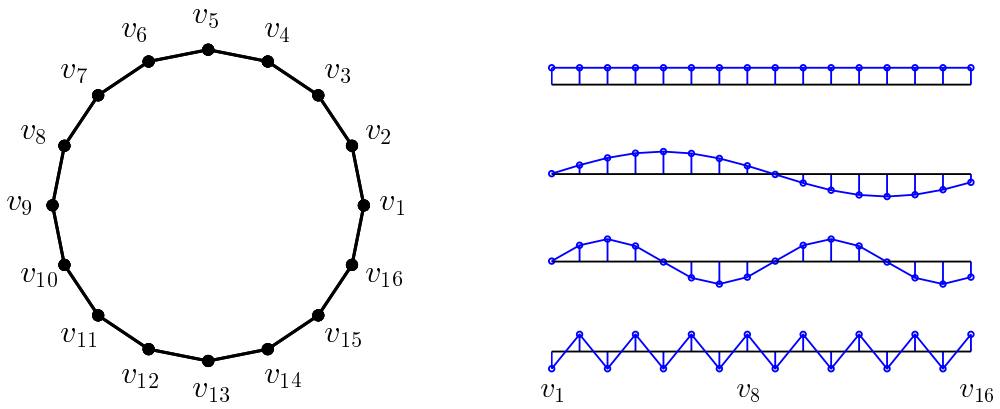
$$\widehat{\boldsymbol{x}} = \boldsymbol{F}^T \boldsymbol{x}, \tag{2.5}$$

where $\widehat{\boldsymbol{x}}$ is called the *graph Fourier transform* of $\boldsymbol{x}$. The matrix $\boldsymbol{F}^T$ represents the Fourier transform operator[1]. Since $\boldsymbol{F}$ is orthogonal, $\boldsymbol{F}\boldsymbol{F}^T = \boldsymbol{I}$. It follows that we can invert the Fourier transform by taking

$$\boldsymbol{x} = \boldsymbol{F}\widehat{\boldsymbol{x}}.$$

Using the Laplacian eigenvectors as a surrogate Fourier basis is a standard approach in the literature for defining signal processing operations on graphs [33, 34, 62, 92, 102, 113, 117]. It may not seem immediately obvious, though, that the analogy is a fair one. In what follows, we provide some justification for this approach.

[1]There may be eigenvalues of $\boldsymbol{L}$ with multiplicity greater than one, so we should really think of the Fourier transform as the set of projections onto the eigenspaces associated with each unique eigenvalue. The Fourier transform defined in this way is unique up to unitary transformations within eigenspaces. We can choose an orthogonal basis in each eigenspace, ensuring that F is orthogonal.

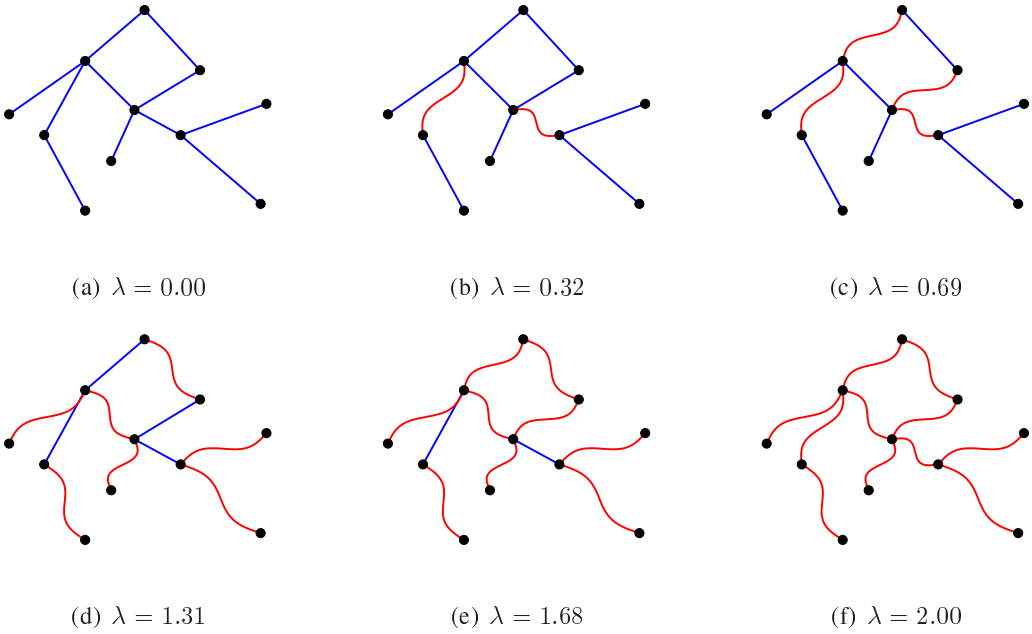


(a) (b)

Figure 2.2: (a) A cycle graph with 16 vertices. Signals defined on this graph are equivalent to standard discrete, periodic signals. (b) Several eigenvectors of the graph Laplacian. These eigenvectors exhibit the sinusoidal characteristics of the DFT basis.

First, consider the special case of a cycle graph, illustrated in Figure 2.2(a). Signals defined on this graph can be thought of as discrete, periodic signals. The Laplacian of this graph is a circulant matrix, and can thus be diagonalized by the real discrete Fourier transform (RDFT) matrix. Thus, in this case the Laplacian eigenbasis is the RDFT basis of sine and cosine functions. Figure 2.2(b) shows several such eigenvectors, which exhibit sinusoidal characteristics with increasing oscillation frequencies. The eigenvalue corresponding to a frequency of $\omega_k = \frac{2\pi k}{N}$ is $1 - \cos(\omega_k)$, which is an increasing function of $|\omega|$ on $[0, \pi]$ (and can be well-approximated by $\frac{\omega^2}{2}$ for small values of k/N.) In a way, signal processing on this graph gives us equivalent results to signal processing of discrete, periodic signals.

For general graphs, of course, the Laplacian eigenbasis is no longer the DFT



(a) $\lambda = 0.00$ (b) $\lambda = 0.32$ (c) $\lambda = 0.69$

(d) $\lambda = 1.31$ (e) $\lambda = 1.68$ (f) $\lambda = 2.00$

Figure 2.3: Some Laplacian eigenvectors of a graph. Straight lines indicate that values on joined vertices have the same sign; wavy lines indicate that there is a sign change between the joined vertices. As is evident, eigenvectors associated with larger eigenvalues correspond to more sign changes and thus faster variation.

basis. Nonetheless, the eigenvectors still satisfy our intuition about frequency. For example, we would like to say that a signal is "highpass" if its value changes significantly between neighboring vertices, and that it is "lowpass" if its value varies very little. To quantify the variation of a signal on a graph, we can construct an $N \times M$ *normalized incidence matrix* $\boldsymbol{S}$ [56], where each column of $\boldsymbol{S}$ corresponds to an edge $e = (u, v)$ and has exactly two nonzero values: $+\frac{1}{\sqrt{\deg(u)}}$ in the row corresponding to vertex u, and $-\frac{1}{\sqrt{\deg(v)}}$ in the row corresponding to vertex v. The choice of (u, v) or (v, u), and therefore the signs involved, is arbitrary for each edge

(though it is important that each column have one positive and one negative value.) For any $\boldsymbol{x} \in \ell^2(G)$, the vector $\boldsymbol{y} = \boldsymbol{S}^T\boldsymbol{x}$ is a signal on the edges of the graph, where each edge has the difference between the normalized values[2] of $\boldsymbol{x}$ on its endpoint vertices. So, in a way, $\boldsymbol{y}$ is the "derivative" of $\boldsymbol{x}$. For any nonzero signal $\boldsymbol{x}$, we can then measure its normalized variation on the graph as

$$\begin{aligned}\frac{1}{\|\boldsymbol{x}\|^2}\sum_{u\sim v}\left(\frac{x(u)}{\sqrt{\deg(u)}}-\frac{x(v)}{\sqrt{\deg(v)}}\right)^2 &= \frac{1}{\|\boldsymbol{x}\|^2}\|\boldsymbol{y}\|^2 \\ &= \frac{1}{\|\boldsymbol{x}\|^2}\boldsymbol{x}^T\boldsymbol{S}\boldsymbol{S}^T\boldsymbol{x} \\ &= \frac{1}{\|\boldsymbol{x}\|^2}\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}, \qquad (2.6)\end{aligned}$$

where the last equality ($\boldsymbol{S}\boldsymbol{S}^T = \boldsymbol{L}$) is well-known and easy to verify [56]. When the signal $\boldsymbol{x}$ is the ith eigenvector $\boldsymbol{f}_i$ of $\boldsymbol{L}$, the normalized variation in (2.6) becomes λ_i, the corresponding eigenvalue. This justifies the usage of Laplacian eigenvalues as frequencies: eigenvectors corresponding to the higher eigenvalues of $\boldsymbol{L}$ are the high-variation components, and the lower eigenvalues correspond to low-variation components. We illustrate this fact with an example in Figure 2.3.

Given the success of multiresolution analysis [39,85,128] in standard signal processing, several authors have considered multiresolution wavelet representations for signals on graphs. Some of these representations rely on the graph Laplacian operator and its analogy to the Fourier transform [34, 62, 92]. In [92], Narang

[2]The normalization by $\frac{1}{\sqrt{\deg(u)}}$ will limit the undue effect on the Laplacian of a vertex with a large number of incident edges.

and Ortega described a natural wish-list for wavelet transforms on graphs: first, they should be invertible and have well-conditioned inverse; second, the transforms should have low redundancies; and finally, they should be localized on the graph and in the spectrum. We are simply interested in quantifying the last, seemingly qualitative desideratum.

2.1.3 Graph and Spectral Spreads

We would like to quantify the localization of a signal on a graph in both the graph and spectral domains. To do so, we look to the definitions of analogous quantities in classical time-frequency analysis. For a nonzero signal $\boldsymbol{x} \in \mathcal{L}^2(\mathbb{R})$, its time spread about a point t_0 is defined by [128]

$$\Delta_{t,t_0}^2 \stackrel{\text{def}}{=} \frac{1}{\|\boldsymbol{x}\|^2} \int_{-\infty}^{\infty} (t - t_0)^2 |x(t)|^2 dt. \tag{2.7}$$

The overall time spread of $x(t)$ is then obtained by minimizing over t_0, *i.e.*,

$$\Delta_t^2 \stackrel{\text{def}}{=} \min_{t_0} \frac{1}{\|\boldsymbol{x}\|^2} \int_{-\infty}^{\infty} (t - t_0)^2 |x(t)|^2 dt, \tag{2.8}$$

where the minimizing value of t_0 is given by $t_0 = \frac{1}{\|\boldsymbol{x}\|^2} \int_{-\infty}^{\infty} t|x(t)|^2 dt$. Generalizing (2.7) to signals defined on graphs, we introduce the following definition [2, 3].

Definition 2.1 (Graph spread)**.** *For a nonzero signal $\boldsymbol{x} \in \ell^2(G)$, its graph spread*

about a vertex u_0 is

$$\Delta^2_{g,u_0}(\boldsymbol{x}) \stackrel{def}{=} \frac{1}{\|\boldsymbol{x}\|^2} \sum_{v \in V} d(u_0, v)^2 x(v)^2 \tag{2.9}$$
$$= \frac{1}{\|\boldsymbol{x}\|^2} \boldsymbol{x}^T \boldsymbol{P}^2_{u_0} \boldsymbol{x},$$

where $d(\cdot, \cdot)$ is the distance metric described in Section 2.1.1, and $\boldsymbol{P}_{u_0}$ is a diagonal matrix defined as

$$\boldsymbol{P}_{u_0} \stackrel{def}{=} \operatorname{diag}\left\{d(u_0, v_1), d(u_0, v_2), \ldots, d(u_0, v_N)\right\}. \tag{2.10}$$

Remark: Similar to (2.8), we can also define the overall (*i.e.*, global) graph spread of $\boldsymbol{x}$ as

$$\Delta^2_g(\boldsymbol{x}) \stackrel{\text{def}}{=} \min_{u_0 \in V} \frac{1}{\|\boldsymbol{x}\|^2} \boldsymbol{x}^T \boldsymbol{P}^2_{u_0} \boldsymbol{x}. \tag{2.11}$$

For our subsequent analysis on uncertainty principles though, we will focus on the local graph spread (*i.e.*, about a particular vertex u_0) as defined in (2.9). Unlike classical domains such as the real line whose topology is shift-invariant, the "landscape" of a graph can look very different around different vertices. Thus, it is important to explicitly specify the center vertex u_0 when considering the graph spread and uncertainty principles. If needed, global versions can always be obtained through finite minimization over all $u_0 \in V$.

The spectral spread of a signal defined on graphs requires more thought. In the classical case, the frequency spread of a real-valued signal $x(t) \in \mathcal{L}^2(\mathbb{R})$ is given by [128]

$$\Delta^2_\omega \stackrel{\text{def}}{=} \frac{1}{\|\boldsymbol{x}\|^2} \int_{-\infty}^{\infty} \omega^2 |\widehat{x}(\omega)|^2 \frac{d\omega}{2\pi}, \tag{2.12}$$

where $\widehat{x}(\omega)$ is the Fourier transform of $x(t)$. This expression is simpler than that of the time spread in (2.7) because the frequency center is chosen to be $\omega_0 = 0$ due to the symmetry of the Fourier transforms of real-valued signals. On recognizing that $\omega^2\widehat{x}(\omega)$ is the Fourier transform of $\frac{-d^2}{dt^2}x(t)$ and using Parseval's identity, we can rewrite (2.12) as

$$\Delta_\omega^2 = \frac{1}{\|\boldsymbol{x}\|^2}\int_{-\infty}^{\infty} x(t)\frac{-d^2}{dt^2}x(t)dt. \tag{2.13}$$

Generalizing to the graph case, treating $\boldsymbol{L}$ as analogous to the operator $-\frac{d^2}{dt^2}$, we obtain the following definition [2, 3].

Definition 2.2 (Spectral spread)**.** *For a nonzero signal $\boldsymbol{x} \in \ell^2(G)$, we define its spectral spread as*

$$\Delta_s^2(\boldsymbol{x}) \stackrel{def}{=} \frac{1}{\|\boldsymbol{x}\|^2}\,\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x} \tag{2.14}$$

$$= \frac{1}{\|\boldsymbol{x}\|^2}\sum_{n=1}^{N}\lambda_n\,|\widehat{x}_n|^2, \tag{2.15}$$

where the second equality follows from the decomposition of $\boldsymbol{L}$ in (2.3) *and the definition of graph Fourier transforms in* (2.5).

Remark: The equivalent definitions in (2.14) and (2.15) reveal two different facets of the spectral spread: while (2.15) perhaps more clearly justifies the "spectral" nature of $\Delta_s^2(\boldsymbol{x})$, the form in (2.14) shows that $\Delta_s^2(\boldsymbol{x})$ can also be understood as the normalized variation of $\boldsymbol{x}$ introduced in (2.6).

2.2 The Uncertainty Principle

Intuitively, we can reason that there should exist a tradeoff between the graph and spectral spreads of a signal. If the graph spread Δ_g^2 is small, then the signal must resemble an impulse centered at some vertex; in this case, the normalized variation (*i.e.*, the spectral spread Δ_s^2) should be high. If instead Δ_s^2 is small, then the signal cannot vary too quickly; it will thus take a long distance for the signal values to drop significantly from the peak value, in which case the graph spread will be high. How can one precisely quantify the above intuition? What are the signals with a given spectral spread that are maximally localized on the graph? These are the fundamental questions addressed in this section.

2.2.1 The Feasibility Region

In the classical uncertainty principle, not all pairs of time-frequency spreads $(\Delta_t^2, \Delta_\omega^2)$ are achievable, and the tradeoff is quantified by the celebrated inequality $\Delta_t^2 \Delta_\omega^2 \geq \frac{1}{4}$, which holds for any nonzero function $x(t) \in \mathcal{L}^2(\mathbb{R})$ [53, 128]. Furthermore, this bound is tight. In fact, any pair of the form $(\Delta_t^2, \Delta_\omega^2) = (c, \frac{1}{4c})$ for $c > 0$ is achievable by a function of the form $x(t) = \exp\left(-\frac{t^2}{4c}\right)$.

In a similar way, we are interested in characterizing the following *feasibility*

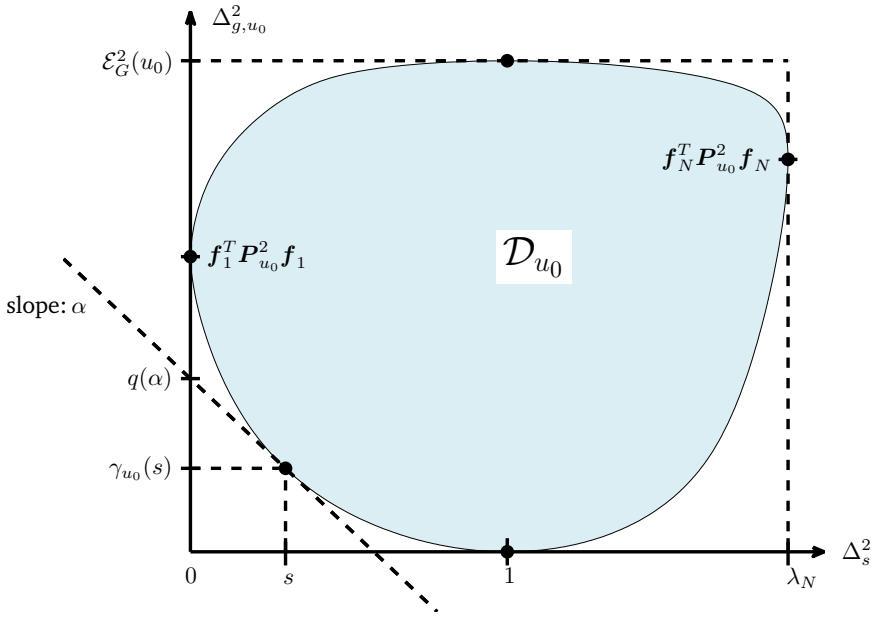


Figure 2.4: The feasibility region $\mathcal{D}_{u_0}$ for the spectral and graph spreads. $\mathcal{D}_{u_0}$ is a bounded and convex set that intersects the horizontal (and vertical) axis at exactly one point. The lower boundary of $\mathcal{D}_{u_0}$ can be implicitly computed by considering supporting lines of varying slopes. The achievable region must lie in the half-plane above the supporting line (found by solving an eigenvalue problem.)

region

$$\mathcal{D}_{u_0} \stackrel{\text{def}}{=} \{(s,g) : \Delta_s^2(\boldsymbol{x}) = s,\ \Delta^2_{g,u_0}(\boldsymbol{x}) = g \quad \text{for some nonzero } \boldsymbol{x} \in \ell^2(G)\}, \tag{2.16}$$

containing all pairs of the form $(\Delta_s^2, \Delta^2_{g,u_0})$ that are achievable on a graph G, using u_0 as the center vertex.

Proposition 2.1. *Let $\mathcal{D}_{u_0}$ be the feasibility region for a* connected *graph G with N vertices. Then the following properties hold:*

(a) $\mathcal{D}_{u_0}$ *is a closed subset of* $[0, \lambda_N] \times [0, \mathcal{E}_G^2(u_0)]$*, where* $\lambda_N \leq 2$ *is the largest eigenvalue of graph Laplacian* $\boldsymbol{L}$*, and* $\mathcal{E}_G(u_0) \stackrel{def}{=} \max_{v \in V} d(u_0, v)$ *is the* eccentricity *of the center vertex* u_0*.*

(b) $\mathcal{D}_{u_0}$ *intersects the horizontal axis at exactly one point,* $(1, 0)$*, and the vertical axis at exactly one point,* $(0, \boldsymbol{f}_1^T \boldsymbol{P}_{u_0}^2 \boldsymbol{f}_1)$*, where* $\boldsymbol{f}_1$ *is the eigenvector defined in* (2.4)*.*

(c) The points $(1, \mathcal{E}_G^2(u_0))$ *and* $(\lambda_N, \boldsymbol{f}_N^T \boldsymbol{P}_{u_0}^2 \boldsymbol{f}_N)$*, where* $\boldsymbol{f}_N$ *is any unit-norm eigenvector associated with* λ_N*, belong to* $\mathcal{D}_{u_0}$*.*

(d) $\mathcal{D}_{u_0}$ *is a convex set if the number of vertices* $N \geq 3$*.*

Proof. (a) The graph and spectral spreads of any nonzero signal can be bounded by the largest and smallest eigenvalues of $\boldsymbol{L}$ and $\boldsymbol{P}_{u_0}^2$. More precisely, using the Rayleigh inequalities [80], we have

$$0 = \lambda_1 \leq \frac{\boldsymbol{x}^T \boldsymbol{L} \boldsymbol{x}}{\boldsymbol{x}^T \boldsymbol{x}} \leq \lambda_N$$

and, similarly,

$$0 \leq \frac{\boldsymbol{x}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x}}{\boldsymbol{x}^T \boldsymbol{x}} \leq \max_{1 \leq i \leq N} (\boldsymbol{P}_{u_0}^2)_{ii} = \mathcal{E}_G^2(u_0).$$

$\mathcal{D}_{u_0}$ is compact, and therefore closed, because it is the image of a compact set under a continuous transform [119]. Specifically, if we take the unit sphere in $\mathbb{R}^N$, a compact set, and apply the map $f : \boldsymbol{x} \mapsto (\Delta_s^2(\boldsymbol{x}), \Delta_{g,u_0}^2(\boldsymbol{x}))$, which is continuous on the unit sphere, we get the whole uncertainty region.

(b) A signal has zero graph spread (*i.e.*, $\Delta_{g,u_0}^2(\boldsymbol{x}) = 0$) if and only if it is an impulse supported on u_0, *i.e.*, $x(v) = c$ if $v = u_0$ and $x(v) = 0$ otherwise, for

some nonzero scalar c. Meanwhile, using (2.14) and (2.6), one can verify that the normalized variation (and thus the spectral spread Δ_s^2) of such impulse signals is equal to 1. It follows that $(1, 0)$ is the only point that lies at the intersection of $\mathcal{D}_{u_0}$ and the horizontal axis. Next, consider the intersection of $\mathcal{D}_{u_0}$ with the vertical axis. Since $\Delta_s^2(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}/\|\boldsymbol{x}\|^2 \geq \lambda_1 = 0$, the spectral spread $\Delta_s^2(\boldsymbol{x}) = 0$ if and only if $\boldsymbol{x}$ is an eigenvector of $\boldsymbol{L}$ associated with the smallest eigenvalue $\lambda_1 = 0$. (See (2.4) for an example.) Such eigenvectors are also unique (up to scalar multiplications) since the smallest eigenvalue λ_1 of connected graphs always has multiplicity one [32].

(c) The inclusion of $(\lambda_N, \boldsymbol{f}_N^T \boldsymbol{P}_{u_0}^2 \boldsymbol{f}_N)$ in $\mathcal{D}_{u_0}$ is clear. For the first point $(1, \mathcal{E}_G^2(u_0))$, consider an impulse function supported at the furthest vertex on the graph from u_0. Similar to (b), we can compute its spectral and graph spreads as $\Delta_s^2 = 1$ and $\Delta_{g,u_0}^2 = \mathcal{E}_G^2(u_0)$, respectively.

(d) See Appendix A.1. □

Remark: Figure 2.4 illustrates a typical feasibility region $\mathcal{D}_{u_0}$ as specified by Proposition 2.1. The boundedness and convexity of $\mathcal{D}_{u_0}$ imply that the entire region can be completely characterized by its upper and lower boundaries: any pair between the two boundaries must also be achievable. Furthermore, the lower boundary must be convex and the upper boundary must be concave.

2.2.2 The Uncertainty Curve

In what follows, we will describe a technique for computing the lower boundary curve of $\mathcal{D}_{u_0}$, which we call the *uncertainty curve*.

Definition 2.3. *Given a connected graph G, the uncertainty curve with respect to a center vertex u_0 is*

$$\begin{aligned}\gamma_{u_0}(s) &\stackrel{def}{=} \min_{\boldsymbol{x}} \Delta^2_{g,u_0}(\boldsymbol{x}) \text{ subject to } \Delta^2_s(\boldsymbol{x}) = s \\ &= \min_{\boldsymbol{x}} \boldsymbol{x}^T \boldsymbol{P}^2_{u_0} \boldsymbol{x} \text{ subject to } \boldsymbol{x}^T\boldsymbol{x} = 1 \text{ and } \boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x} = s, \end{aligned} \tag{2.17}$$

for all $s \in [0, \lambda_N]$.

Remark: We could also define and study the upper boundary curve of $\mathcal{D}_{u_0}$ in a similar way. We choose to focus on the lower boundary curve because it provides an uncertainty bound analogous to the classical bound (2.1). We will say that a signal $\boldsymbol{x}$ achieves the uncertainty curve if $\Delta^2_{g,u_0}(\boldsymbol{x}) = \gamma_{u_0}(\Delta_s(\boldsymbol{x})^2)$.

We note that (2.17) is a quadratically constrained quadratic program [22]. The equality constraints make the problem nonconvex. On differentiating the corresponding Lagrangian function

$$\Lambda(\boldsymbol{x}; \alpha, \lambda) \stackrel{\text{def}}{=} \boldsymbol{x}^T \boldsymbol{P}^2_{u_0} \boldsymbol{x} - \alpha(\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x} - s) - \lambda(\boldsymbol{x}^T\boldsymbol{x} - 1),$$

we see that the optimal solution $\boldsymbol{x}^*$ to (2.17) must satisfy

$$(\boldsymbol{P}^2_{u_0} - \alpha\boldsymbol{L})\boldsymbol{x}^* = \lambda\boldsymbol{x}^*$$

for some $\alpha, \lambda \in \mathbb{R}$. If we treat α as being fixed, then the above equality becomes an eigenvalue problem. This observation leads us to study the matrix-valued function

$$\boldsymbol{M}(\alpha) \stackrel{\text{def}}{=} \boldsymbol{P}_{u_0}^2 - \alpha \boldsymbol{L}. \tag{2.18}$$

For any α, the smallest eigenvalue of $\boldsymbol{M}(\alpha)$, denoted by

$$q(\alpha) \stackrel{\text{def}}{=} \lambda_{\min}(\boldsymbol{M}(\alpha)),$$

and its associated eigenspace, denoted by $\mathcal{S}(\alpha)$, are key to our analysis of the uncertainty curve $\gamma_{u_0}(s)$.

Proposition 2.2. *For any $\alpha \in \mathbb{R}$ and any unit-norm eigenvector $\boldsymbol{v}$ in $\mathcal{S}(\alpha)$, the point $(\boldsymbol{v}^T \boldsymbol{L} \boldsymbol{v}, \boldsymbol{v}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{v})$ is on $\gamma_{u_0}(s)$.*

Proof. Let $\boldsymbol{x}$ be an arbitrary signal with $\|\boldsymbol{x}\| = 1$. By definition, $\Delta_{g,u_0}^2(\boldsymbol{x}) - \alpha \Delta_s^2(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{M}(\alpha) \boldsymbol{x}$. Applying Rayleigh's inequality to $\boldsymbol{M}(\alpha)$ thus leads to

$$\Delta_{g,u_0}^2(\boldsymbol{x}) - \alpha \Delta_s^2(\boldsymbol{x}) \geq q(\alpha) \tag{2.19}$$

$$= \boldsymbol{v}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{v} - \alpha \boldsymbol{v}^T \boldsymbol{L} \boldsymbol{v}, \tag{2.20}$$

where (2.20) comes from the fact that $\boldsymbol{v}$ is an eigenvector associated with $q(\alpha)$. Let $s = \boldsymbol{v}^T \boldsymbol{L} \boldsymbol{v}$. On specializing the relationship (2.20) to those signals $\boldsymbol{x}$ satisfying $\Delta_s^2(\boldsymbol{x}) = s$, we have

$$\Delta_{g,u_0}^2(\boldsymbol{x}) \geq \boldsymbol{v}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{v},$$

which indicates that the point $(\boldsymbol{v}^T \boldsymbol{L} \boldsymbol{v}, \boldsymbol{v}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{v})$ must lie on the uncertainty curve $\gamma_{u_0}(s)$. □

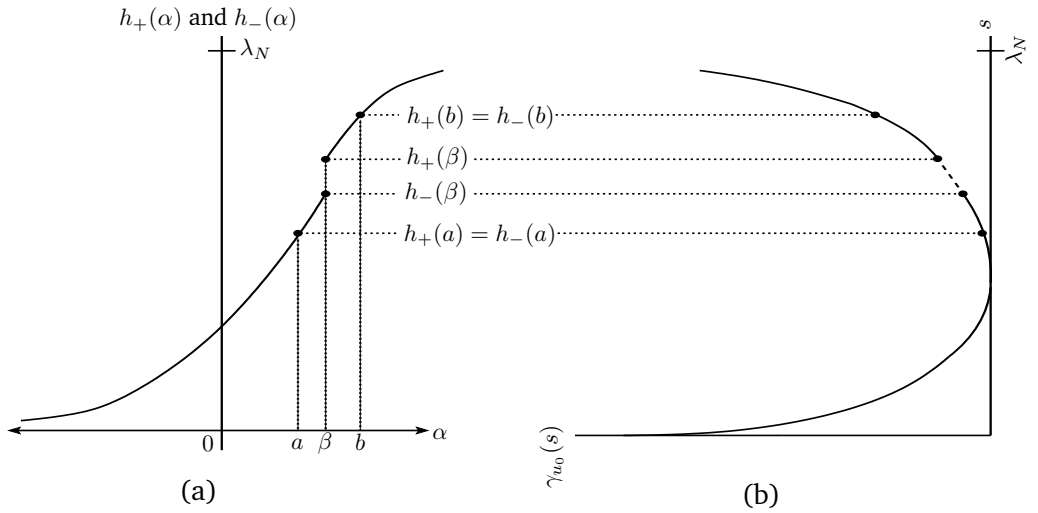


Figure 2.5: The mapping of the eigenvectors in $S(\alpha)$ onto the s–g plane is shown. In (a), $h_+(\alpha)$ and $h_-(\alpha)$ are plotted against α (they coincide except at jumps in the plot.) They are, respectively, the maximum and minimum spectral spreads of elements of the eigenspace $S(\alpha)$. Any element of $S(\alpha)$ determines a point on the graph of $\gamma_{u_0}(s)$. When $S(\alpha)$ is of dimension greater than one, it corresponds to a line segment on $\gamma_{u_0}(s)$.

There is an interesting geometric interpretation of the above derivations: as illustrated in Figure 2.4, for any α, the inequality in (2.19) defines a half-plane in which $\mathcal{D}_{u_0}$ must lie. The boundary of the half-plane, a line of slope α defined by

$$\Delta_{g,u_0}^2 - \alpha\Delta_s^2 = q(\alpha),$$

provides a tight lower bound to $\mathcal{D}_{u_0}$. Varying the values of α generates a family of such half-planes, the intersection of which contains $\mathcal{D}_{u_0}$. For readers familiar with convex analysis, we note that $q(\alpha)$ is the Legendre transform of $\gamma_{u_0}(s)$ [22].

Proposition 2.2 guarantees that any nonzero eigenvector of $\boldsymbol{M}(\alpha)$ associated with the smallest eigenvalue $q(\alpha)$ generates a point on the curve $\gamma_{u_0}(s)$. Next, we

will show that the converse is also true: every point on $\gamma_{u_0}(s)$ is achieved by an eigenvector in $\mathcal{S}(\alpha)$ for some α. To establish this result, we need to introduce the following two functions:

$$h_+(\alpha) \stackrel{\text{def}}{=} \max_{\boldsymbol{x}\in\mathcal{S}(\alpha):\,\|\boldsymbol{x}\|=1} \boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x} \tag{2.21}$$

$$h_-(\alpha) \stackrel{\text{def}}{=} \min_{\boldsymbol{x}\in\mathcal{S}(\alpha):\,\|\boldsymbol{x}\|=1} \boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x},$$

which measure, respectively, the maximum and minimum spectral spread (*i.e.*, the horizontal coordinate on the s–g plane) that can be achieved by eigenvectors in $\mathcal{S}(\alpha)$.

Lemma 2.1. *The following properties hold for $h_+(\alpha)$ and $h_-(\alpha)$.*

(a) They are increasing functions, i.e.*, $h_+(\alpha_1) \le h_+(\alpha_2)$ and $h_-(\alpha_1) \le h_-(\alpha_2)$ for all $\alpha_1 < \alpha_2$.*

(b) They have the same limits as $|\alpha|$ tends to infinity:

$$\lim_{\alpha\to-\infty} h_+(\alpha) = \lim_{\alpha\to-\infty} h_-(\alpha) = 0, \tag{2.22}$$

and

$$\lim_{\alpha\to+\infty} h_+(\alpha) = \lim_{\alpha\to+\infty} h_-(\alpha) = \lambda_N. \tag{2.23}$$

(c) On any finite interval $[a,b]$, the functions $h_+(\alpha)$ and $h_-(\alpha)$ differ on at most a finite number of points, denoted by $\mathcal{B} \stackrel{\text{def}}{=} \{\beta_1,\beta_2,\ldots,\beta_k\}$ for some $k \ge 0$. Except for these points, $h_+(\alpha)$ and $h_-(\alpha)$ coincide, are continuous, and satisfy

$$h_+(\alpha) = h_-(\alpha) = -q'(\alpha), \qquad \text{for all } \alpha \in [a,b]\setminus\mathcal{B}, \tag{2.24}$$

where $q'(\alpha)$ is the derivative of $q(\alpha)$. At the points, if any, where they do differ, $h_+(\alpha)$ and $h_-(\alpha)$ have jump discontinuities. Moreover, for all $\beta \in \mathcal{B}$,

$$h_+(\beta) = \lim_{\alpha \to \beta^+} h_+(\alpha) > \lim_{\alpha \to \beta^-} h_-(\alpha) = h_-(\beta),$$

where the limits are taken as α approaches β from the positive and negative sides, respectively.

Proof. See Appendix A.2. □

The results of Lemma 2.1 are illustrated in Figure 2.5(a), where we plot a typical example of $h_+(\alpha)$ and $h_-(\alpha)$: as α increases from $-\infty$ to $+\infty$, the values of the functions increase from 0 to λ_N. Within any finite interval, $h_+(\alpha) = h_-(\alpha)$ except at a finite number of points (*e.g.*, the point β in the figure). At these "jump points", $h_+(\alpha)$ is right-continuous, whereas $h_-(\alpha)$ is left-continuous.

Since we are only considering connected graphs, $\lambda_1 = 0$ has multiplicity 1, and so $\boldsymbol{f}_1$ is the unique vector (up to scaling) that achieves the uncertainty curve with $\Delta_s^2 = 0$. At the other end, λ_N may have multiplicity, but some vector in its eigenspace will achieve the uncertainty curve with $\Delta_s^2 = \lambda_N$. For values of $s \in (0, \lambda_{\max})$, we can use the following theorem to precisely characterize vectors that achieve the uncertainty curve at s.

Theorem 2.1. *A signal $\boldsymbol{x} \in \ell^2(G)$ with $\Delta_s^2(\boldsymbol{x}) \in (0, \lambda_{\max})$ achieves the uncertainty curve,* i.e.*, $\Delta_{g,u_0}^2(\boldsymbol{x}) = \gamma(\Delta_s^2(\boldsymbol{x}))$,* if and only if *it is a nonzero eigenvector in $\mathcal{S}(\alpha)$ for some α.*

Proof. The "if" direction has been established in Proposition 2.2. To prove the "only if" direction, we will show that for any signal $\boldsymbol{x} \in \ell^2(G)$ that achieves the uncertainty curve, there is an α and a unit-norm eigenvector $\boldsymbol{v} \in \mathcal{S}(\alpha)$ such that $\boldsymbol{v}^T\boldsymbol{L}\boldsymbol{v} = \Delta_s^2(\boldsymbol{x})$. Since both $\boldsymbol{x}$ and $\boldsymbol{v}$ lie on the uncertainty curve (with the former given as an assumption and the latter guaranteed by Proposition 2.2), we have $\Delta_{g,u_0}^2(\boldsymbol{x}) = \boldsymbol{v}^T\boldsymbol{P}_{u_0}^2\boldsymbol{v}$, and thus

$$\frac{1}{\|\boldsymbol{x}\|^2}\,\boldsymbol{x}^T\boldsymbol{M}(\alpha)\boldsymbol{x} = \Delta_{g,u_0}^2(\boldsymbol{x}) - \alpha\Delta_s^2(\boldsymbol{x}) = \boldsymbol{v}^T\boldsymbol{M}(\alpha)\boldsymbol{v} = q(\alpha).$$

Now, since $q(\alpha)$ is the smallest eigenvalue of $\boldsymbol{M}(\alpha)$, the equality above implies that $\boldsymbol{x}$ must also be an eigenvector associated with $q(\alpha)$. In fact, $\boldsymbol{x}$ will be equal to $\boldsymbol{v}$ (up to a scalar multiple) if $q(\alpha)$ has multiplicity one. The remainder of the proof verifies the claim, namely, for any $s \in (0, \lambda_N)$ we can find an α and a unit-norm eigenvector $\boldsymbol{v} \in \mathcal{S}(\alpha)$ such that $\boldsymbol{v}^T\boldsymbol{L}\boldsymbol{v} = s$.

By part (b) of Lemma 2.1, we can always find some $a < b$ such that $h_-(a) < s < h_+(b)$. Furthermore, part (c) of Lemma 2.1 ensures that, within the interval $[a, b]$, the two functions $h_+(\alpha)$ and $h_-(\alpha)$ differ (and are discontinuous) on at most a finite number of points. For notational simplicity, and without loss of generality, we assume that there is only one such discontinuity point, denoted by $\beta \in [a, b]$. As shown in Figure 2.5, the interval $[h_-(a), h_+(b)]$ can now be written as the union of three subintervals

$$[h_-(a), h_-(\beta)),\ [h_-(\beta), h_+(\beta)],\ \text{and}\ (h_+(\beta), h_+(b)],$$

to one of which s must belong.

We first consider the case where $s \in [h_-(a), h_-(\beta))$. Lemma 2.1 says that $h_-(\alpha)$ is a continuous function on $[a, \beta]$. By the intermediate value theorem, there exists some $\alpha_0 \in [a, \beta]$ such that $h_-(\alpha_0) = s$. By definition, $h_-(\alpha_0) = \min_{\boldsymbol{z} \in \mathcal{S}(\alpha_0):\, \|\boldsymbol{z}\|=1} \boldsymbol{z}^T \boldsymbol{L} \boldsymbol{z}$. Since the eigenspace $\mathcal{S}(\alpha_0)$ has finite dimensions, the minimization can always be achieved by some unit-norm eigenvector $\boldsymbol{v} \in \mathcal{S}(\alpha_0)$, *i.e.*, $s = h_-(\alpha_0) = \boldsymbol{v}^T \boldsymbol{L} \boldsymbol{v}$. The same line of reasoning can be used when s belongs to the third subinterval, $(h_+(\beta), h_+(b)]$. This leaves us with the remaining case when $s \in [h_-(\beta), h_+(\beta)]$. Let

$$\boldsymbol{v}_+ \stackrel{\text{def}}{=} \operatorname*{argmax}_{\boldsymbol{z} \in \mathcal{S}(\beta):\|\boldsymbol{z}\|=1} \boldsymbol{z}^T \boldsymbol{L} \boldsymbol{z} \quad \text{and} \quad \boldsymbol{v}_- \stackrel{\text{def}}{=} \operatorname*{argmin}_{\boldsymbol{z} \in \mathcal{S}(\beta):\|\boldsymbol{z}\|=1} \boldsymbol{z}^T \boldsymbol{L} \boldsymbol{z},$$

and consider the vector-valued function $\boldsymbol{y}(\theta) \stackrel{\text{def}}{=} \frac{\cos(\theta)\boldsymbol{v}_+ + \sin(\theta)\boldsymbol{v}_-}{1+\sin(2\theta)\boldsymbol{v}_+^T \boldsymbol{v}_-}$, defined for $\theta \in [0, \pi/2]$. The denominator is nonzero for every θ, since $\boldsymbol{v}_- \neq -\boldsymbol{v}_+$ [otherwise we would have $h_-(\beta) = h_+(\beta)$]. So $\boldsymbol{y}(\theta)$ is of unit norm and is a continuous function of θ. It also must belong to $\mathcal{S}(\beta)$ since it is a linear combination of two elements of the subspace. Furthermore, $\boldsymbol{y}(0)^T \boldsymbol{L} \boldsymbol{y}(0) = h_+(\beta)$ and $\boldsymbol{y}(\pi/2)^T \boldsymbol{L} \boldsymbol{y}(\pi/2) = h_-(\beta)$. By the intermediate value theorem, $\boldsymbol{y}(\theta)$ for $\theta \in [0, \pi/2]$ achieves all the values in between. In particular, there exists some θ_0 such that $\boldsymbol{y}(\theta_0)^T \boldsymbol{L} \boldsymbol{y}(\theta_0) = s$. We note that since every element of $\mathcal{S}(\beta)$ achieves a point on the line $g - \beta s = q(\beta)$, this interpolation procedure amounts to including the straight line segment between the two endpoints as part of the uncertainty curve. □

Remark: If $\mathcal{S}(\alpha)$ is one-dimensional for every $\alpha \in [a, b]$, or more generally if

there is a single distinct eigenvalue function that achieves the minimum on $[a, b]$, then from Theorem 2.1 as well as Lemma 2.1 and its proof, $q(\alpha)$ is analytic on $[a, b]$ and the corresponding portion of the uncertainty curve can be expressed in parametric form as

$$\begin{cases} s(\alpha) = -q'(\alpha) \\ \\ \gamma_{u_0}(s) = q(\alpha) - \alpha q'(\alpha), \end{cases} \tag{2.25}$$

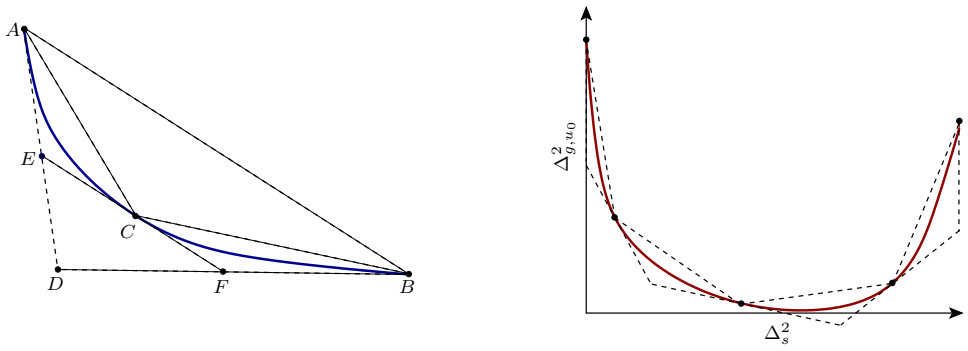
where the first equality is due to (2.24) and the second is due to the fact that any vector in $\mathcal{S}(\alpha)$ must achieve a point on the line $g - \alpha s = q(\alpha)$.

In general, Theorem 2.1 and its proof justify a way to obtain the uncertainty curve: for every α, we find the eigenvectors associated with the smallest eigenvalue of $\boldsymbol{M}(\alpha)$. These eigenvectors will give us points on $\gamma_{u_0}(s)$. By "sweeping" the values of α from $-\infty$ to ∞, the entire curve can then be traced.

2.2.3 Fast Approximation Algorithm

In practice, of course, we must sample and work with a finite set of α's, which lead to an approximation of the true curve. In what follows, we describe an efficient algorithm that can compute an approximation—more specifically, an upper and lower bound—of $\gamma_{u_0}(s)$ with any desired accuracy.

Since $\gamma_{u_0}(s)$ is the lower boundary of the convex region $\mathcal{D}_{u_0}$, it is itself a convex function. We can therefore use the *sandwich* algorithm introduced in [111] and presented here as Algorithm 2.1 to approximate it. The algorithm can be easily un-



(a) (b)

Figure 2.6: An illustration of the *sandwich* algorithm. (a) A single refinement step on a segment of the uncertainty curve. (b) Two refinement steps on the full curve.

derstood by studying Figure 2.6(a): consider a segment of the curve $\gamma_{u_0}(s)$ with two end points A and B, whose coordinates are denoted by $(a, \gamma_{u_0}(a))$ and $(b, \gamma_{u_0}(b))$, respectively. Also given are supporting lines[3] containing the end points, represented by the line segments AD and BD. Due to the convexity of $\gamma_{u_0}(s)$, the chord that connects A to B must lie entirely above the curve and thus form an upper bound. Similarly, the combination of AD and BD forms a piecewise linear lower bound of $\gamma_{u_0}(s)$.

To refine these two initial bounds, let α be the slope of the chord, *i.e.*,

$$\alpha = \frac{\gamma_{u_0}(b) - \gamma_{u_0}(a)}{b - a}. \tag{2.26}$$

Computing the smallest eigenvalue $q(\alpha)$ and the associated eigenvectors of $\boldsymbol{M}(\alpha)$, we can obtain a new point on the curve, denoted by C in Figure 2.6(a). The s-

[3]A supporting line is a line that intersects a curve but does not separate any two points on the curve [22].

g coordinates of C are $(\boldsymbol{v}^T\boldsymbol{L}\boldsymbol{v}, \boldsymbol{v}^T\boldsymbol{P}_{u_0}^2\boldsymbol{v})$, where $\boldsymbol{v}$ is a unit-norm element in the eigenspace $\mathcal{S}(\alpha)$. Our previous analysis in Section 2.2.2—in particular, (2.19) and (2.20)—guarantees that the line

$$g - \alpha s = q(\alpha),$$

which passes through C, must be a supporting line of $\gamma_{u_0}(s)$. In other words, α is a subderivative of $\gamma_{u_0}(s)$ at point C, and is the derivative if it exists. This property, together with the construction of α in (2.26), also ensures that C is always located between A and B. As illustrated in the figure, the curve is now bounded above by joining the three points (A, C and B), and it is bounded below by joining the three supporting lines (AE, EF and FB).

The above procedure can then be repeated, in a recursive fashion, on the two curve segments AC and CB. Each stage of the recursion roughly doubles the number of points in the approximation, and we proceed until a fixed number of refinements have been computed. Figure 2.6(b) shows the lower and upper bounds of $\gamma_{u_0}(s)$ obtained by starting from two initial points $(0, \boldsymbol{f}_1^T\boldsymbol{P}_{u_0}^2\boldsymbol{f}_1)$ and $(\lambda_N, \boldsymbol{f}_N^T\boldsymbol{P}_{u_0}^2\boldsymbol{f}_N)$ and running the algorithm for two refinement iterations, involving a total of five eigenvalue evaluations (each corresponding to a single point drawn on the curve.) We can see that the proposed algorithm starts producing reasonable approximations of $\gamma_{u_0}(s)$ after just a small number of steps.

Let $\eta_u^{(n)}(\cdot)$ and $\eta_\ell^{(n)}(\cdot)$ denote, respectively, the upper and lower bounds the algorithm generates after n eigenvalue evaluations. We measure the quality of approx-

Algorithm 2.1 Approximate $\gamma_{u_0}(s)$ curve (sandwich algorithm.)

Require: $\boldsymbol{L}$, $\boldsymbol{P}_{u_0}^2$, desired number of points N

Ensure: Upper bound curve $\gamma_{u_0}^u(s)$; Lower bound curve $\gamma_{u_0}^\ell(s)$

$p_1 \leftarrow (0, \boldsymbol{x}_0^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x}_0)$; $\quad p_2 \leftarrow (\lambda_{\max}, \boldsymbol{x}_N^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x}_N)$

$t_1 \leftarrow (\infty, 0)$; $\quad t_2 \leftarrow (\infty, \lambda_{\max})$

while $\#\mathcal{P} < N$ **do**

  $i \leftarrow 1$

  while $i < \#\mathcal{P}$ **do**

    $\alpha \leftarrow -\frac{s_{i+1} - s_i}{g_{i+1} - g_i}$

    $(\boldsymbol{x}, q) = \texttt{min-eig}(\boldsymbol{P}_{u_0}^2 + \alpha \boldsymbol{L})$ {Note: $\|\boldsymbol{x}\| = 1$}

    $s \leftarrow \boldsymbol{x}^T \boldsymbol{L} \boldsymbol{x}$

    $g \leftarrow \boldsymbol{x}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x}$

    $p \leftarrow \texttt{concat}[p_{1:i} \quad (s, g) \quad p_{i+1:\text{end}}]$

    $t \leftarrow \texttt{concat}[t_{1:i} \quad (\alpha, q) \quad t_{i+1:\text{end}}]$

    $i \leftarrow i + 2$

  end while

end while

$\gamma_{u_0}^u(\cdot) \leftarrow \texttt{connect-dots}(p)$

$\gamma_{u_0}^\ell(\cdot) \leftarrow \texttt{connect-lines}(t)$

imation by computing the Hausdorff distance [111] between these two bounds, defined as

$$d(n) = \sup_{s_1} \inf_{s_2} \left[(s_1 - s_2)^2 + (\eta_u^{(n)}(s_1) - \eta_\ell^{(n)}(s_2))^2 \right]^{\frac{1}{2}}.$$

Informally, the Hausdorff distance $d(n)$ is small if the two bounding curves are close to each other. The following theorem, which follows directly from [111, Theorem 3], shows that $d(n)$ is of order $1/n^2$.

Theorem 2.2. *Let $\varepsilon > 0$ be any preset precision level. To get $d(n) \leq \varepsilon$, it is sufficient to run the approximation algorithm until we have $n \geq \max\left\{4, \sqrt{9W/\varepsilon} + 2\right\}$, where $W = \sqrt{\lambda_N^2 + \mathcal{E}_G^4(u_0)}$.*

Remark: In many practical applications, the underlying graph G is large but sparse. Correspondingly, $\boldsymbol{M}(\cdot)$ are sparse matrices. Obtaining an approximation of $\gamma_{u_0}(s)$ within a given precision ε then boils down to computing (*e.g.*, via iterative power methods) the smallest eigenvalue and an associated eigenvector of about $\mathcal{O}(1/\sqrt{\varepsilon})$ sparse matrices.

Instead of approximating the whole curve, we may wish to find $\gamma_{u_0}(s)$ only for some particular value of s, as well as the signal that achieves it. The sandwich algorithm can be modified slightly to this end. At each step of the approximation procedure, we can choose to refine only the segment containing s, ignoring all other segments. Iterating in this way, we will find both $\gamma_{u_0}(s)$ and the vector with spectral spread s that achieves the bound.

It may be that we are not interested in approximating the whole curve, but we wish to find $\gamma_{u_0}(s)$ and the vector that achieves it for a particular value of s. Therefore we will also describe a different scheme (Algorithm 2.2) that effectively computes the uncertainty curve at a single point.

Algorithm 2.2 Approximate $\gamma_{u_0}(s)$

Require: $n \geq 0 \vee x \neq 0$
Ensure: $y = x^n$
  $y \Leftarrow 1$
  if $n < 0$ **then**
    $X \Leftarrow 1/x$
    $N \Leftarrow -n$
  else
    $X \Leftarrow x$
    $N \Leftarrow n$
  end if

2.3 The Uncertainty Curve for Special Graph Families

The uncertainty curves for several standard graph families are analyzed in this section. The structure and regularity of complete graphs and star graphs make it possible to find closed-form expressions for their corresponding curves. For Erdős-Rényi random graphs [50, 51], we will derive and compute analytical approximations for the expected (*i.e.*, mean) curves under different parameters. Throughout this section, the distance metric $d(\cdot, \cdot)$ is assumed to be the geodesic distance.

2.3.1 Complete Graphs

A complete graph, illustrated in Figure 2.7(a), is a fully-connected graph in which every pair of distinct vertices is connected by an edge [79]. It is often used to model fully-connected subgraphs, or *cliques*, in real-world networks [96]. The Laplacian matrix of a complete graph with N vertices is given by

$$\boldsymbol{L}_{ij} = \begin{cases} 1, & \text{if } i = j; \\ -\frac{1}{N-1}, & \text{otherwise}, \end{cases} \tag{2.27}$$

i.e., the diagonal of $\boldsymbol{L}$ is all 1, and the off-diagonal elements are all equal to $-\frac{1}{N-1}$. It is easy to verify that $\boldsymbol{L}$ has eigenvalue 0 with multiplicity 1, and eigenvalue $\frac{N}{N-1}$ with multiplicity $N-1$. Without loss of generality, we can choose the first vertex as the center. The diagonal distance matrix is then

$$\boldsymbol{P}_{u_0} = \mathrm{diag}\{0, 1, 1, \ldots, 1\}. \tag{2.28}$$

We would like to compute the uncertainty curve $\gamma(s)$ for a complete graph for $s \in [0, \frac{N}{N-1}]$. First, we will show that any vector that achieves the uncertainty curve has a special form.

Proposition 2.3. *For a complete graph, suppose $\widetilde{\boldsymbol{x}}$ achieves the uncertainty curve. Then $\widetilde{\boldsymbol{x}}$ is of the form*

$$\widetilde{\boldsymbol{x}} = [x_1, x_2, x_2, \ldots, x_2]^T. \tag{2.29}$$

Proof. See Appendix A.3. □

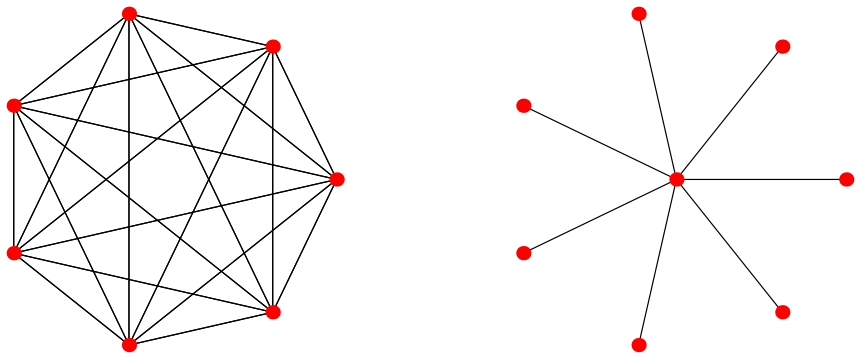


Figure 2.7: (a) A complete graph is one in which every vertex is connected to every other vertex. (b) A star graph includes a single central vertex connected by a single edge to each other vertex, and no other edges

The result in Proposition 2.3 suggests that, for complete graphs, we need only consider vectors of the form in (2.29). Enforcing the unit-norm constraint on (2.29), we can further simplify these eigenvectors as

$$\widetilde{\boldsymbol{x}}(\theta) = [\cos\theta, \frac{\sin\theta}{\sqrt{N-1}}, \frac{\sin\theta}{\sqrt{N-1}}, \ldots, \frac{\sin\theta}{\sqrt{N-1}}]^T$$

for some parameter θ. The graph spread in this case is given by

$$\Delta^2_{g,u_0} = \sum_{i=1}^{N-1} 1 \cdot \frac{\sin^2\theta}{N-1} = \frac{1}{2} - \frac{1}{2}\cos 2\theta,$$

where the second equality is due to a standard trigonometric identity. Meanwhile,

by using the variational form in (2.6), we can compute the spectral spread as

$$\begin{aligned}\Delta_s^2 &= (N-1)\left(\frac{\cos\theta}{\sqrt{N-1}} - \frac{\sin\theta}{N-1}\right)^2 \\ &= \frac{N}{2N-2} - \frac{1}{\sqrt{N-1}}\sin 2\theta + \frac{N-2}{2N-2}\cos 2\theta. \end{aligned} \tag{2.30}$$

Combining these two expressions and using the identity $\sin^2 2\theta + \cos^2 2\theta = 1$, we can see that the uncertainty curve $\gamma_{u_0}(s)$ is part of the ellipse given by

$$(2\Delta_{g,u_0}^2 - 1)^2 + (N-1)\left(\Delta_s^2 + \frac{N-2}{N-1}\Delta_{g,u_0}^2 - 1\right)^2 = 1. \tag{2.31}$$

For fixed $s = \Delta_s^2$, solving for $\gamma_{u_0}(s) = \Delta_{g,u_0}^2$ [by picking the smaller of the two solutions to (2.31)] leads to

$$\begin{aligned}&\gamma_{u_0}(s) = \\ &\frac{N - s(N-2) - 2\sqrt{1-(N-2)(s-1)-(N-1)(s-1)^2}}{4 + (N-2)^2/(N-1)},\end{aligned} \tag{2.32}$$

for $0 \leq s \leq \frac{N}{N-1}$. Thus, the curve is the entire lower half of the ellipse given by (2.31). When the graph is large (*i.e.*, $N \gg 1$), this curve converges to a straight line $\gamma_{u_0}(s) = 1 - s$ in the s–g plane.

2.3.2 Star Graphs

A star graph [56] with N vertices has one central vertex and $N-1$ leaves, each connected by a single edge to the center. Illustrated in Figure 2.7(b), it is a prototypical example of a hub in a network [96]. The Laplacian matrix can be

expressed in block form as

$$\boldsymbol{L} = \begin{pmatrix} 1 & -\frac{1}{\sqrt{N-1}}\mathbf{1}_{N-1}^T \\ -\frac{1}{\sqrt{N-1}}\mathbf{1}_{N-1} & \boldsymbol{I}_{N-1} \end{pmatrix}, \tag{2.33}$$

where $\mathbf{1}_{N-1}$ is the $(N-1)$-vector of all ones, and $\boldsymbol{I}_{N-1}$ is the $(N-1)\times(N-1)$ identity matrix. Since the graph is bipartite, the largest eigenvalue of $\boldsymbol{L}$ is always equal to 2 [32]. Let u_0 be the center of the star; the diagonal distance matrix is again given by $\boldsymbol{P}_{u_0} = \mathrm{diag}\{0,1,1,\ldots,1\}$.

Just as for the complete graph, we can always represent signals that achieve the uncertainty curve on star graphs as $\widetilde{\boldsymbol{x}}(\theta) = [\cos\theta, \frac{\sin\theta}{\sqrt{N-1}}, \frac{\sin\theta}{\sqrt{N-1}}, \ldots, \frac{\sin\theta}{\sqrt{N-1}}]^T$ for some θ (see the remark in Appendix A.3 for justification). Now, the graph spread is given by $\Delta_{g,u_0}^2 = \sin^2\theta = \frac{1}{2} - \frac{1}{2}\cos 2\theta$; again, by using (2.6), the spectral spread can be computed as

$$\begin{aligned}\Delta_s^2 &= (N-1)\left(\frac{\cos\theta}{\sqrt{N-1}} - \frac{\sin\theta}{\sqrt{N-1}}\right)^2 \\ &= 1 - \sin 2\theta.\end{aligned}$$

The lower bound curve is thus the lower part of the ellipse defined by

$$\left(\Delta_s^2 - 1\right)^2 + (2\Delta_{g,u_0}^2 - 1)^2 = 1.$$

Written explicitly, the curve is

$$\gamma_{u_0}(s) = \frac{1}{2}\left(1 - \sqrt{s(2-s)}\right), \quad \text{for } 0 \le s \le 2. \tag{2.34}$$

We note that, unlike the complete graph case, this curve does not depend on the size of the graph.

2.3.3 Erdős-Rényi Random Graphs

An Erdős-Rényi random graph G is generated by taking N vertices and selecting each pair of vertices to be an edge with probability p, independent of all other potential edges. We denote by $\mathcal{G}_p(N,p)$ the statistical ensemble of the resulting graphs. First studied by Erdős and Rényi [50, 51], $\mathcal{G}_p(N,p)$ may be the simplest random graph model. Although they do not capture all of the behaviors of real networks, Erdős-Rényi graphs are an excellent theoretical model because they lend themselves to tractable analysis.

To study the properties of the uncertainty curves for Erdős-Rényi graphs, we generated several realizations drawn from $\mathcal{G}_p(N,p)$ and used the approximation algorithm described in Section 2.2.3 to compute their uncertainty curves. It quickly emerged that the curves for different realizations generated with the same parameters were, for reasonable sizes of N, tightly clustered around a common mean curve. This is illustrated in Figure 2.8, which shows the mean curves and estimated standard deviations for several parameter values. In what follows, we develop an analytic approximation for computing the expected (*i.e.* mean) uncertainty curve for different choices of parameters N and p.

Recall from the definition of the uncertainty curve that we are trying to approximate the expectation of

$$\gamma_{u_0}(s) = \min_{\boldsymbol{x}\in\ell^2(G)} \boldsymbol{x}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x} \text{ subject to } \|\boldsymbol{x}\|^2 = 1 \text{ and } \boldsymbol{x}^T \boldsymbol{L} \boldsymbol{x} = s \tag{2.35}$$

over random graphs drawn from $\mathcal{G}_p(N,p)$. The matrices $\boldsymbol{P}_{u_0}^2$ and $\boldsymbol{L}$ and the optimal vector $\boldsymbol{x}$ that solves the minimization problem are all random quantities. Since $\gamma_{u_0}(s)$ is obtained through a nonconvex quadratic program, there is generally no closed-form expressions linking $\gamma_{u_0}(s)$ to $\boldsymbol{P}_{u_0}^2$ and $\boldsymbol{L}$. As a result, directly computing the expectation of $\gamma_{u_0}(s)$ will be difficult. To make the problem tractable, we proceed by replacing $\boldsymbol{x}^T\boldsymbol{P}_{u_0}^2\boldsymbol{x}$ and $\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ in (2.35) with their respective expected values and minimizing after the fact. Later we will see that this strategy turns out to be very effective in generating accurate approximations.

Another observation that emerged from our numerical experiment was a characteristic of the vectors that achieved the bound with $s \leq 1$: these vectors were all approximately radial functions, *i.e.*, the value at any vertex v was a function of $d(u_0,v)$. Because this simplifies the analysis greatly, we will only consider the part of the curve with $s \leq 1$, which corresponds to signals that are maximally localized in both the graph and spectral domains. We will explicitly incorporate this assumption by focusing on vectors whose values depend only on distance from u_0. In this case, the original N-dimensional vector $\boldsymbol{x} \in \ell^2(G)$ can be represented by a smaller vector $\boldsymbol{y}$, with $x(v) = y(d(u_0,v))$. The dimensionality of $\boldsymbol{y}$ is equal to $\mathcal{E}_G(u_0)+1$, where $\mathcal{E}_G(u_0)$ is the eccentricity of the center vertex. We note that $\mathcal{E}_G(u_0)$ is a random variable that in principle can take any value between 0 and $N-1$. When N is large, however, we can find a small number $d_{\max} \sim \mathcal{O}(\log N)$ such that $\mathcal{E}_G(u_0) \leq d_{\max}$ with high probability [10]. So, in what follows, we will treat $\boldsymbol{y}$ as a vector in $\mathbb{R}^{d_{\max}+1}$.

For a given, deterministic $\boldsymbol{y}$, we will compute the expectations (over the randomness of the graph model) of $\|\boldsymbol{x}\|^2$ and $\boldsymbol{x}^T\boldsymbol{P}_{u_0}^2\boldsymbol{x}$. To that end, we define f_d as the probability that a vertex v chosen uniformly at random from $V\backslash\{u_0\}$ has a distance $d(u_0, v) = d$. The special case $f_1 = p$ is easy to verify. For the other cases, we will use the results of Blondel *et al.* [19], who developed a recursive formula[4] to find (approximate) analytical expressions of the entire sequence $\{f_d\}$. The expected number of vertices at a distance $d \geq 1$ is $(N-1)f_d$. It follows that, for fixed $\boldsymbol{y}$,

$$\begin{aligned}\mathbb{E}\left[\|\boldsymbol{x}\|^2\right] &= \mathbb{E}\left[\sum_{v\in V} y(d(u_0,v))^2\right] \\ &\approx y^2(0) + \sum_{k=1}^{d_{\max}} (N-1) f_k y^2(k) \end{aligned} \tag{2.36}$$

and

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{x}^T\boldsymbol{P}_{u_0}\boldsymbol{x}\right] &= \mathbb{E}\left[\sum_{v\in V} d(u_0,v)^2 x(v)^2\right] \\ &\approx \sum_{k=1}^{d_{\max}} k^2 (N-1) f_k y^2(k), \end{aligned} \tag{2.37}$$

where the approximations are due to the truncation of $\boldsymbol{y}$ at dimension $d_{\max}$.

The spectral spread is more complicated. We start with the expression

$$\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x} = \sum_{u\sim v}\left(\frac{x(u)}{\sqrt{\deg(u)}} - \frac{x(v)}{\sqrt{\deg(v)}}\right)^2.$$

By assuming that the degree of every vertex is approximately equal to its expecta-

[4]Unlike our construction, they allowed v to be any vertex in V, including u_0; thus, in their result, $f_0 = \frac{1}{N}$, and all other values of f_d differ from ours by a factor of $\frac{N-1}{N}$. For large N the difference is negligible.

tion $(N-1)p$, we write

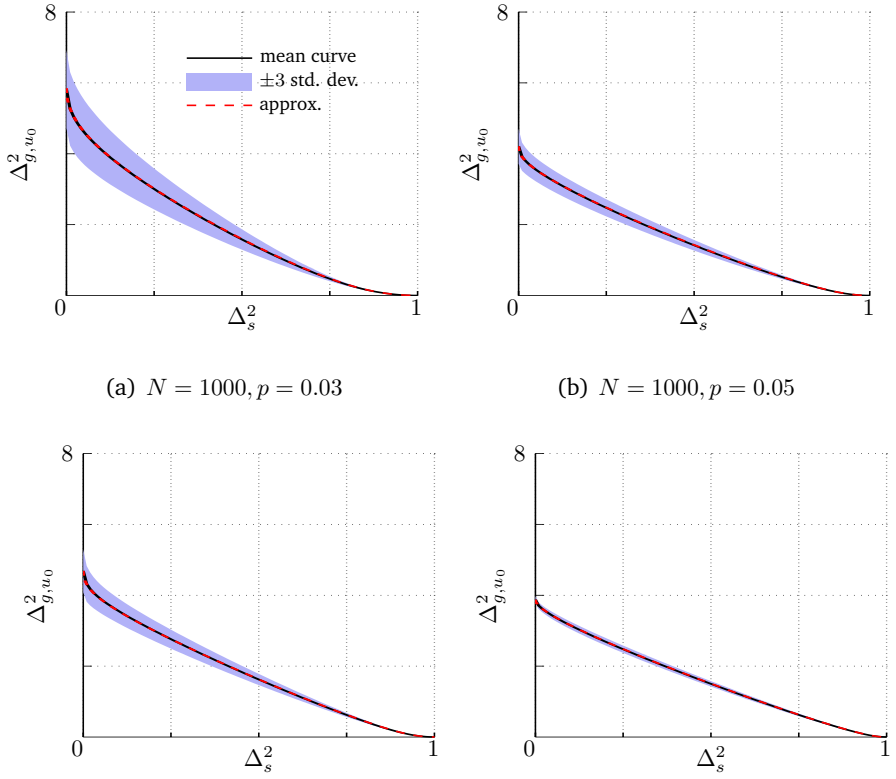
$$\boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x} \approx \frac{1}{(N-1)p} \sum_{u \sim v} (x(u) - x(v))^2. \tag{2.38}$$

Recall that $x(v) = y(d(u_0, v))$. Consequently, the only edges that contribute to (2.38) are those between vertices at *different* distances from u_0. Since a vertex at distance k can only be connected to vertices at a distance of $k-1$ and $k+1$, we simply need to characterize $M_{k,k+1}$, the expected number of edges from vertices at a distance k to vertices at a distance $k+1$, for $k = 0$ to $d_{\max} - 1$. The expected value of the spectral spread can then be obtained as

$$\mathbb{E}\left[\boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}\right] \approx \frac{1}{(N-1)p} \sum_{k=0}^{d_{\max}-1} M_{k,k+1} \big(y(k+1) - y(k)\big)^2. \tag{2.39}$$

It is easy to see that $M_{0,1} = (N-1)p$, since that is simply the expected number of edges incident upon u_0. The other terms of $M_{k,k+1}$ can be approximated through a recurrence relation. First, we observe that the expected number of vertices at distance k is $(N-1)f_k$ and the expected number of vertices *not* at distance k (not counting u_0) is $(N-1)(1-f_k)$. Thus, we can approximate the total number of *potential* edges between these two disjoint sets of vertices is $(N-1)^2 f_k(1-f_k)$. Since each potential edge will be chosen with probability p, we get that $M_{k-1,k} + M_{k,k+1} \approx (N-1)^2 p f_k(1-f_k)$, which leads to the following approximate recurrence relation

$$\begin{cases} M_{0,1} = (N-1)p \\ \\ M_{k,k+1} \approx (N-1)^2 p f_k(1-f_k) - M_{k-1,k}, \quad \text{for } k \geq 1. \end{cases} \tag{2.40}$$



(a) $N = 1000, p = 0.03$

(b) $N = 1000, p = 0.05$

(c) $N = 2000, p = 0.03$

(d) $N = 2000, p = 0.05$

Figure 2.8: Uncertainty curves for Erdős-Rényi graphs. For each choice of (N, p) parameters, 1000 Erdős-Rényi graphs were generated and their uncertainty curves for $s \in [0, 1]$ were computed using the sandwich approximation procedure described in Section 2.2. The geodesic distance function is used. Each curve was interpolated to generate comparable curves on a regular grid. For each parameter choice, the mean and standard deviation of the interpolated curve was computed over the ensemble. The mean curve is plotted on the graphs as a solid line, with shaded areas illustrating the three standard deviation levels. Meanwhile, the approximate expected value computed before generating the curves is plotted as a dashed red line. The shape of the uncertainty curve is clearly quite stable across each ensemble, especially as N and p increase, and the approximate expectation curve is quite accurate.

The expressions in (2.36), (2.37), and (2.39) show that the expected values of the squared norm, graph spread, and spectral spread are all nonnegative quadratic forms involving the vector $\boldsymbol{y} \in \mathbb{R}^{d_{\max}+1}$. It follows that we can write

$$\mathbb{E}\left[\|\boldsymbol{x}\|^2\right] \approx \boldsymbol{y}^T \boldsymbol{H}_a \boldsymbol{y}, \;\; \mathbb{E}\left[\boldsymbol{x}^T \boldsymbol{P}_{u_0} \boldsymbol{x}\right] \approx \boldsymbol{y}^T \boldsymbol{P}_a^2 \boldsymbol{y}, \text{ and} \tag{2.41}$$
$$\mathbb{E}\left[\boldsymbol{x}^T \boldsymbol{P}_{u_0} \boldsymbol{x}\right] \approx \boldsymbol{y}^T \boldsymbol{L}_a \boldsymbol{y},$$

for some positive semidefinite matrices $\boldsymbol{H}_a, \boldsymbol{P}_a^2, \boldsymbol{L}_a$, respectively. Substituting these expectations for their (random) counterparts in (2.35), we compute our approximation of the expected uncertainty curve, $\tilde{\gamma}_{u_0}(s)$, as

$$\tilde{\gamma}_{u_0}(s) = \tag{2.42}$$
$$\min_{\boldsymbol{y} \in \mathbb{R}^{d_{\max}+1}} \boldsymbol{y}^T \boldsymbol{P}_a^2 \boldsymbol{y} \text{ subject to } \boldsymbol{y}^T \boldsymbol{H}_a \boldsymbol{y} = 1 \text{ and } \boldsymbol{y}^T \boldsymbol{L}_a \boldsymbol{y} = s.$$

We note that this minimization problem (a quadratic program with quadratic constraints) has exactly the same mathematical structure as the one previously studied in (2.17). Using the same techniques derived in Section 2.2.2, we can show that any solution to (2.42) satisfies the (generalized) eigenvalue problem

$$(\boldsymbol{P}_a^2 - \alpha \boldsymbol{L}_a)\boldsymbol{y} = \tau_{\min}(\alpha) \boldsymbol{H}_a \boldsymbol{y} \tag{2.43}$$

for some value of α, where $\tau_{\min}(\alpha)$ is the smallest (generalized) eigenvalue. As before, we can construct a sandwich approximation to the curve by solving (2.43) for a sequence of α's.

Despite the various approximations made along the way, the analytical solution obtained in (2.42) fits experiment remarkably well. As illustrated in Figure 2.8,

the resulting analytic curves (shown in dashed lines) match almost perfectly with the observed sample average (shown in solid lines). We note that the matrices in (2.42) are of size $d_{\max} \times d_{\max}$, which is much smaller than $N \times N$. For example, for the $\mathcal{G}_p(10^6, 10^{-4})$ model, we would have $d_{\max} = 4$ (the smallest d such that $1 - \sum_{k=1}^{d} f_k < 10^{-7}$.)

Thus, the analytic approximation derived here can be computed far faster than the actual uncertainty curve for any realization of the model, and does not itself require any realization to be generated.

2.4 Diffusion Processes and Uncertainty Bounds

In constructing dictionaries to represent signals on graphs, one would like the dictionary elements to be localized in both graph and spectral domains. Quantifying the signal localization in these two domains and studying their fundamental tradeoff have been one of the motivations of this work. To test the theoretical results and the computational algorithm presented in Section 2.2, we consider two graph wavelet transforms in the literature: the diffusion wavelets of Coifman and Maggioni [34] and the spectral graph wavelet transform of Hammond *et al.* [62]. The localization properties of these two constructions are studied on a graph visualized in Figure 2.9(a) based on the network of football games played in the 2000

regular season by NCAA Division I-A[5] teams [55]. While the spectral graph wavelet transform does not downsample the graph, the diffusion wavelet transform does. In our experiment, the center vertex u_0 is chosen to be one of the vertices that remain in the downsampled graph at the coarsest level of the diffusion wavelet transform.

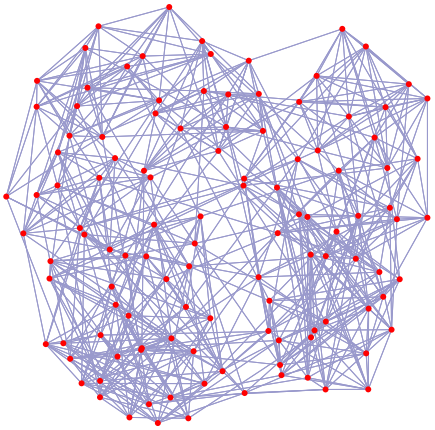
Figure 2.9(b) shows several scaling functions from both constructions plotted against the uncertainty curve $\gamma_{u_0}(s)$, with the latter obtained by using the sandwich algorithm in Section 2.2.3. In this and all subsequent experiments, we use eight refinement iterations (for a total of 257 sparse eigenvalue evaluations) to plot the uncertainty curves. At this level, we find the lower and upper approximations of $\gamma_{u_0}(s)$ to be visually indistinguishable. As predicted, both the spectral graph wavelet and diffusion wavelet constructions result in basis elements that obey the computed bound. In fact, they follow the curve quite well.

The diffusion wavelets are based on the evolution of a discrete time diffusion process on a graph. In the classical setting, where the signal domain is the real line, there is a strong connection between the *continuous time* diffusion process and the Heisenberg uncertainty curve: to see this, consider a diffusion (*i.e.* heat) equation

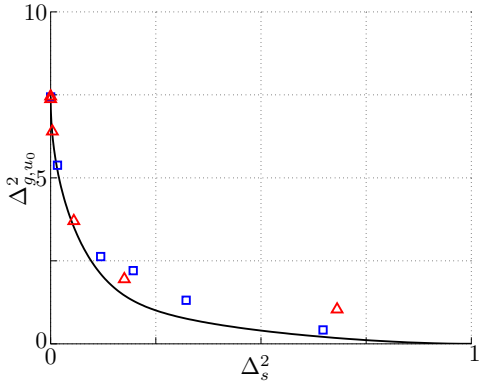
$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial y^2}, \tag{2.44}$$

where $u(y,t)$ is a function of $y, t \in \mathbb{R}$. This equation governs the conduction of heat in physical processes, and its solution was the original motivation for Fourier analysis. The fundamental solution to (2.44), *i.e.*, the solution with the initial condition

[5]Now the Football Bowl Subdivision (FBS).



(a)



(b)

Figure 2.9: (a) Network of football games between NCAA Division I-A teams in the 2000 regular season [55]; (b) Spectral spread versus graph spread on this graph. Solid line: computed uncertainty curve $\gamma_{u_0}(s)$. Triangles: scaling functions in diffusion wavelets [34]. Squares: scaling functions in spectral graph wavelet transform (SGWT) [62]. (The true SGWT scaling functions are not related to the wavelet functions by a two-scale relation; here, we simply take the cumulative sum of the coarsest-level scaling function and higher-level wavelet functions.)

that $u(y,0) = \delta(y - y_0)$ for a given y_0, is the Gaussian kernel

$$K(t, y, y_0) = \frac{1}{\sqrt{4\pi t}} e^{-\frac{(y-y_0)^2}{4t}}.$$

Thus, if we start with an impulse and evolve according to (2.44), at time t we get a function with time spread t and frequency spread $\frac{1}{4t}$, achieving the classical Heisenberg uncertainty $\Delta_t^2 \, \Delta_\omega^2 \geq \frac{1}{4}$ with equality. In other words, the diffusion kernels on the real line are exactly the signals that achieve the time-frequency uncertainty bound.

This line of thought motivated us to consider a continuous-time diffusion process on graphs, governed by an equation analogous to (2.44):

$$\frac{d\boldsymbol{x}}{dt} = -\boldsymbol{L}\boldsymbol{x}, \tag{2.45}$$

where $\boldsymbol{L}$ is the graph Laplacian. With the initial condition $\boldsymbol{x}(0) = \boldsymbol{\delta}_{u_0}$, the solution to (2.45) is [33]

$$\boldsymbol{x}(t) = e^{-t\boldsymbol{L}}\boldsymbol{\delta}_{u_0} = \sum_{i=1}^{N} e^{-t\lambda_i} \boldsymbol{f}_i \boldsymbol{f}_i^T \boldsymbol{\delta}_{u_0}, \tag{2.46}$$

where $e^{-t\boldsymbol{L}}$ is the matrix exponential of $\boldsymbol{L}$, $\{\lambda_i\}$ are the eigenvalues of $\boldsymbol{L}$, and $\{\boldsymbol{f}_i\}$ are the corresponding eigenvectors. Denote by $\eta_{u_0}(s)$ the curve in the s–g plane traced out by the diffusion process. The curve can be given in parametric form as

$$\begin{cases} s(t) = \frac{\boldsymbol{x}(t)^T \boldsymbol{L} \boldsymbol{x}(t)}{||\boldsymbol{x}(t)||^2} \\ \eta_{u_0}(s) = \frac{\boldsymbol{x}(t)^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x}(t)}{||\boldsymbol{x}(t)||^2}. \end{cases} \tag{2.47}$$

We show in Appendix A.5 that $s(t)$ is a strictly decreasing function of t; therefore it is one-to-one. Furthermore, $s(0) = 1$ and $\lim_{t\to\infty} s(t) = 0$. All together, this guarantees that the function $\eta_{u_0}(s)$ is well-defined for every $s \in (0, 1]$.

We plot in Figure 2.10 the diffusion curve $\eta_{u_0}(s)$ and the uncertainty curve $\gamma_{u_0}(s)$ for three different graphs: a random geometric graph [101] that can capture the connectivity of wireless sensor networks; an unstructured triangular mesh[6] for finite element analysis [118]; and a small-world graph [131] that serves as the mathematical model for social and various other empirical networks. The geodesic distance function is used. In all three cases, the spreads of the diffusion process, though not exactly achieving the bounds as in the classical setting, match the uncertainty curves remarkably well.

The following proposition, proved in Appendix A.4, asserts that for certain special graphs the match between $\eta_{u_0}(s)$ and $\gamma_{u_0}(s)$ is exact.

Proposition 2.4. *For all $s \in (0, 1]$, $\eta_{u_0}(s) = \gamma_{u_0}(s)$ if (a) G is a complete graph with N vertices and u_0 is any vertex; or (b) G is a star graph with N vertices and u_0 is the vertex with degree $N - 1$.*

For general graphs we can show that, under certain conditions, the low-order derivatives of the uncertainty curve and the diffusion curve match.

[6]This graph was generated using the Mesh2D MATLAB toolbox written by Darren Engwirda, available online at MATLAB Central (`http://www.mathworks.com/matlabcentral/fileexchange/25555`).

Proposition 2.5. *Let G be any connected graph and u_0 be any vertex on G. Then* $\eta_{u_0}(1) = \gamma_{u_0}(1) = 0$, $\left.\frac{d\eta_{u_0}}{ds}\right|_{s=1} = \left.\frac{d\gamma_{u_0}}{ds}\right|_{s=1} = 0$, *and*

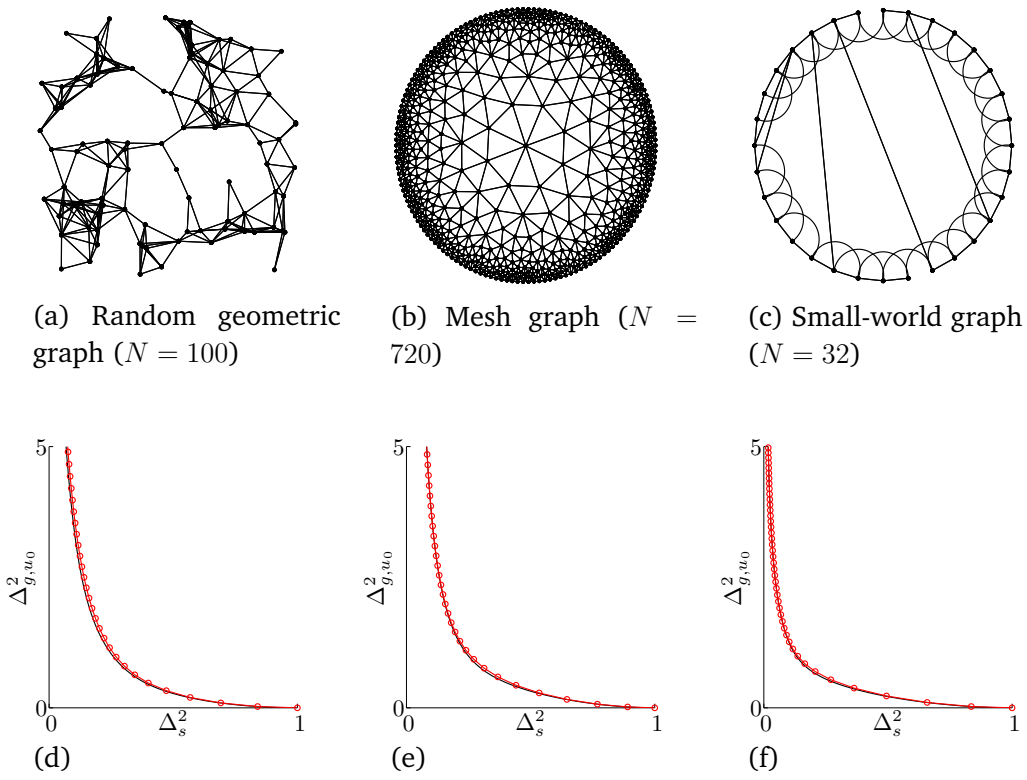
$$\left.\frac{d^2\gamma_{u_0}}{ds^2}\right|_{s=1} = \frac{\deg u_0}{2\sum_{v\sim u_0}\frac{1}{d(v,u_0)^2\deg v}} \leq \left.\frac{d^2\eta_{u_0}}{ds^2}\right|_{s=1}$$
$$= \frac{\deg u_0}{2}\frac{\sum_{v\sim u_0}\frac{d(v,u_0)^2}{\deg v}}{\left(\sum_{v\sim u_0}\frac{1}{\deg v}\right)^2}, \tag{2.48}$$

with equality if and only if $d(v, u_0)$ *is identical for every* $v \sim u_0$.

This proposition is proved in Appendix A.5. It is easy to verify that the geodesic distance satisfies the condition required for equality in (2.48). Extrapolating the observations in Figure 2.10 and results in Propositions 2.4 and 2.5 leads us to believe that diffusion kernels on arbitrary graphs will always be close to optimal in graph and spectral localizations. We leave further rigorous study of this tantalizing conjecture as an important line of future work.

2.5 Summary

In this chapter, we developed an uncertainty principle for signals defined on graphs, analogous to the classical Heisenberg uncertainty principle in time-frequency analysis. After presenting quantitative definitions of the signal "spreads" in the graph and spectral domains, we provided a complete characterization of the feasibility region achieved by these two quantities. The lower boundary of the region, which is analogous to the classical uncertainty bound (2.1), was shown to be



(a) Random geometric graph ($N = 100$)

(b) Mesh graph ($N = 720$)

(c) Small-world graph ($N = 32$)

(d)

(e)

(f)

Figure 2.10: Diffusion process versus the uncertainty curve for three types of graph. (a) A random geometric graph [101], (b) a triangular mesh [118], and (c) a small-world graph [131]. Below each graph, (d), (e), and (f) show the associated uncertainty curves (solid black line). A continuous-time diffusion process is run on each graph, beginning with an impulse at one vertex, and the resulting spreads are plotted (solid red line with circles). The circles are evenly spaced in time. The diffusion process tracks the curve closely, though close examination reveals that the match is not exact.

achieved by eigenvectors associated with the smallest eigenvalues of a particular matrix-valued function. Furthermore, the convexity of the uncertainty curve allows it to be efficiently approximated by solving a sequence of eigenvalue problems. We derived closed-form formulas of the uncertainty curves for complete graphs and star graphs, and developed a fast analytical approximation for the expected uncertainty curve for Erdős-Rényi random graphs. The localization properties of two existing wavelet transforms were evaluated. Finally, numerical experiments and analytical results led us to an intriguing connection between diffusion processes on graphs and the uncertainty bounds.

Chapter 3

Optimal Detection of Random Walks on Graphs

SUPPOSE WE WISH to make sense of a sequence of observations from nodes in a graph. The observations form a spatiotemporal matrix, where each column contains the measurements at all nodes at a particular snapshot in time. As illustrated in Figure 3.1, we need to distinguish between two hypotheses: (a) every observation is just meaningless zero-mean Gaussian noise, or (b) an agent is undergoing a random walk on the graph and the measurement at its location at each time has an elevated mean. We do not know the exact path of the agent, but we do know its dynamics: with the graph structure assumed known, the agent's movements follow a well-defined finite-state Markov chain. In effect, we would like to exploit our knowledge of the graph structure (or the Markov chain) to help detect

a possibly very weak signal.

In practice, this problem can arise from the detection of an intruder via a sensor network; the motion of a potential intruder might be modeled as a random walk on a graph representing the network, and one is tasked with testing the hypothesis that an intruder is currently present based on noisy measurements from each sensor. This kind of model has also been used in the detection of frequency-hopping or other highly oscillatory signals [28]. More generally, it can be interpreted as the detection of a hidden Markov process, a problem with many applications (see, *e.g.*, [81, 90, 112, 121, 124].)

The task we have is a kind of *combinatorial testing* problem [1, 11, 45, 68], in that there is an exponentially large number of paths that could be anomalous. Thus, the alternative hypothesis is in fact a composite of an exponentially large number of simple hypotheses. Despite this complexity, the optimal Neyman-Pearson detector in our problem turns out to be easy to derive and computationally tractable. However, its performance is not so simple to characterize.

We will use the (type-II) error exponent, which measures the rate of decay of the miss detection probability when the false alarm probability is held fixed, as the performance metric. One should expect it to depend on the signal-to-noise ratio (SNR) *and* the degree to which the Markov dynamics restrict the paths of the agent. If the SNR is too low, the true path will not be very different from the noise. But if the number of potential paths is very small, it may be easy to rule out

false alarms, and performance will be better than when the number is very high. As the main focus of this chapter, we will characterize the error exponent of the optimal detector and quantify the above intuition. We do this by deriving a fully rigorous lower bound to the error exponent, using ideas borrowed from statistical physics [87–89, 97].

Related and Prior Work

Detecting a continuous Gauss-Markov process in Gaussian noise is a classical signal processing problem that has been extensively studied (see, *e.g.*, [72, 114].) Hypothesis testing that tries to distinguish between two different finite-state Markov chains based on *noiseless* realizations is also well-understood [74, 82, 91]. In this work, we focus on the related problem of detecting random walks on directed and weighted graphs (which are finite state Markov chains) based on noisy observations that are perturbed by additive Gaussian noise. These observations neither satisfy the Markov property nor are jointly Gaussian, making the problem a more difficult one.

There is some prior work on detecting hidden Markov processes such as the one we consider in this chapter. The structure of the optimal detector for a finite-state Markov chain in noise was addressed in [123, 124]. We are interested in going further and characterizing the asymptotic performance of the optimal detector by computing the error exponent. For the Gauss-Markov case, a closed-form expres-

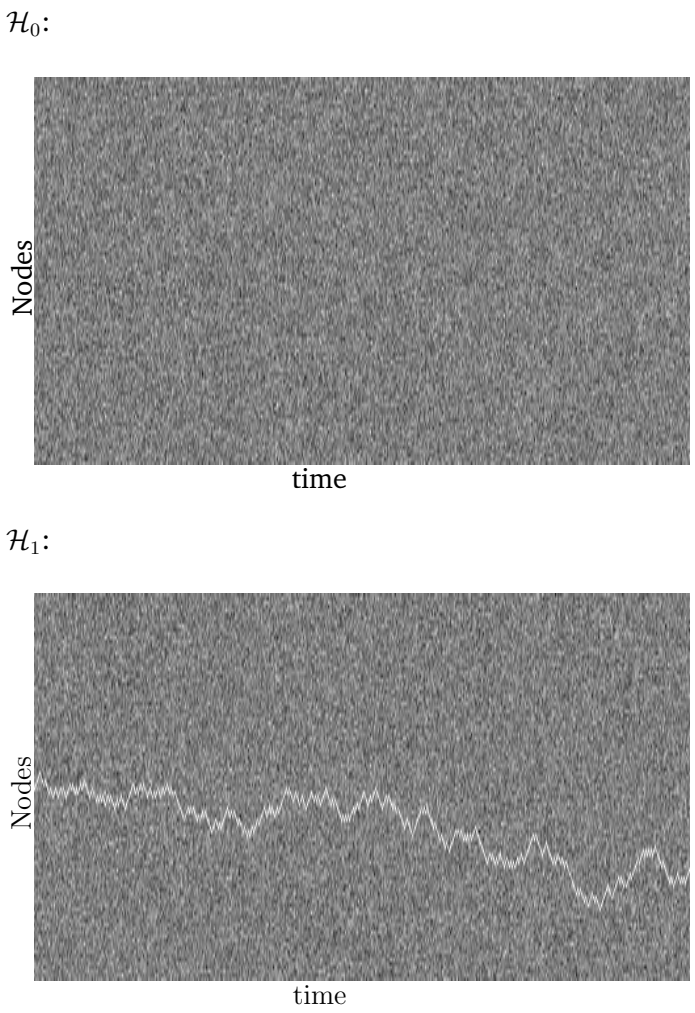


Figure 3.1: Illustration of the two hypotheses under consideration. Each column of the observation matrix shows the measurements at all nodes at a particular point in time. The null hypothesis $\mathcal{H}_0$ (top) is that all of the measurements are just noise. The alternate hypothesis $\mathcal{H}_1$ (bottom) is that a single node has an elevated mean at each time, and that node is chosen by a random walk. Here, we have illustrated a random walk on a line graph, but in this chapter we consider the general case of any finite-state Markov chain.

sion for the error exponent was derived by Sung *et al.* [121] using a state space representation. Our problem turns out to be more challenging. The error exponent, we shall see, is equal to the top Lyapunov exponent of the product of a sequence of random matrices [37, 54], a problem known to be difficult [126]. Leong *et al.* [81] described a numerical technique to approximately compute the error exponent for detecting a two-state Markov chain in noise by discretizing a certain integral equation. Unfortunately, numerical solutions based on discretization become computationally intractable for general Markov chains with a large number of states, the case we address in this chapter. In principle, one can always use Monte Carlo simulations to estimate the Lyapunov exponent (and thus the error exponent.) However, they will not easily provide insights relating the error exponents to the SNR and the Markov chain structures.

Finally, we note that our problem is closely related to the general task of detecting nonzero-mean components of a Gaussian random vector [12, 45, 68]. Addario-Berry *et al.* characterized the performance in a very general setting [1], bounding the Bayesian risk of the test; but in that work all of the nonzero-mean support sets under test are equiprobable and there is no Markov structure. Arias-Castro *et al.* considered a problem similar to ours where a path on a graph has elevated mean while all other nodes are zero-mean Gaussians [11]; instead of a time series, they considered a single snapshot in the asymptotic regime of very large graphs.

In this chapter, we consider general graphs (or Markov chains) with an arbi-

trary number of nodes. Drawing upon techniques originally developed in statistical physics [87–89,97], we compute a lower bound on the error exponent that appears in practice to be quite sharp. The lower bound exhibits a *phase transition* at a certain threshold SNR, separating the detectable and undetectable regimes. Some of these results were previously presented in [5,8], but we only justified them through nonrigorous arguments common in the statistical physics literature. In this chapter we use large deviations theory [41,48] to provide a fully rigorous derivation for the lower bound.

Contributions

We will precisely formulate the hypothesis testing problem in Section 3.1, and introduce and motivate the error exponent as the performance metric. The main contributions of the chapter will follow:

(1) In Section 3.2, we prove that the error exponent for this problem is well-defined and equal to the asymptotic Kullback-Leibler (KL) divergence rate of the two hypotheses. We do this by generalizing the standard Chernoff-Stein lemma [41], which gives the error exponent for independent and identically distributed (i.i.d.) hypotheses, to the Markovian case.

(2) Later in Section 3.2, we develop upper and lower bounds for the error exponent. The upper bound is a simple genie bound. The lower bound is derived

borrowing techniques from statistical physics—it is related to the *free energy density* of a new "spin glass" model [87–89, 97, 122].

(3) We show how to explicitly compute the statistical physics-based lower bound. A rigorous proof of the expression is technical, so we present our results in two steps: first, we provide in Section 3.3 a high-level overview of our approach, emphasizing ideas and intuitions rather than rigor. Our discussions there also serve as a roadmap to the various results in Section 3.4, where we use large deviations theory to rigorously derive an expression for the lower bound and show how to compute it parametrically. The lower bound we derive exhibits a phase transition at an SNR equal to twice the entropy rate of the Markov chain. Below the threshold SNR, the bound is exactly equal to zero, indicating poor performance; above the threshold, there is rapid improvement in performance as the SNR increases.

(4) In Section 3.4.4, we compare the true error exponent (as estimated via Monte Carlo simulations) to the lower bound and find that the bound fully captures its behavior, which appears to undergo a smoothed version of the phase transition at the predicted threshold. In the detectable SNR regime (above the threshold), our bound is also far better than an alternative bound obtained by ignoring the Markov structure, especially when the graph size is large.

We offer some concluding remarks in Section 3.5.

3.1 Problem Formulation

We consider testing the two hypotheses illustrated in Figure 3.1. The data form a matrix $\boldsymbol{Y}^N = [y_{m,n}]$ with $1 \leq m \leq M$ and $1 \leq n \leq N$, where M is the number of nodes in the graph and N is the number of observation times. As we allow the graph to be directed and weighted, the dynamics of an agent following a random walk on the graph can model any finite-state Markov chain. The two hypotheses are as follows:

$$\mathcal{H}_0 : y_{m,n} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$

$$\mathcal{H}_1 : \boldsymbol{s} = (s_1, s_2, \ldots, s_N) \sim \text{Markov}(\boldsymbol{P})$$

$$y_{m,n}|\boldsymbol{s} \overset{\text{indep.}}{\sim} \begin{cases} \mathcal{N}(\beta,1), & \text{if } m = s_n \\ \mathcal{N}(0,1), & \text{if } m \neq s_n, \end{cases}$$

where $\boldsymbol{P}$ is the *known* transition matrix of an irreducible and aperiodic M-state Markov chain [so that $\Pr(s_{n+1} = j | s_n = i) = p_{i,j}$, the ijth entry of $\boldsymbol{P}$].

Under the null hypothesis $\mathcal{H}_0$, the measurements are just i.i.d. zero-mean standard Gaussian noise. Under the alternate hypothesis $\mathcal{H}_1$, there is a sequence of states $\boldsymbol{s} = (s_1, s_2, \ldots, s_N) \in \{1, \ldots, M\}^N$ produced by a Markov chain with transition matrix $\boldsymbol{P}$, and we assume that s_1 is drawn from its unique stationary distribution $\boldsymbol{\pi}$. By the Perron-Frobenius theorem for irreducible matrices [67], the elements of $\boldsymbol{\pi}$ are all positive, meaning each state has a positive probability of being initially chosen. Given the state sequence $\boldsymbol{s}$, the entries of the data matrix $\boldsymbol{Y}^N$ are still inde-

pendent Gaussian random variables. The difference is just that, in each column n the Gaussian random variable at the s_nth entry has an elevated mean β. This can be interpreted as the "signature" or "evidence" left behind by the agent. The variance in both hypotheses is set to 1 without loss of generality; what matters is the signal to noise ratio (SNR) of β^2. In what follows, we will use $P_0(\cdot)$ and $P_1(\cdot)$ to refer to the probability laws under $\mathcal{H}_0$ and $\mathcal{H}_1$, respectively, and $\mathbb{E}_0$ and $\mathbb{E}_1$ to refer to the corresponding expectation operators.

The optimal detector, that which minimizes the miss detection probability for a fixed false alarm probability, is the Neyman-Pearson detector [36]. The corresponding decision rule compares the likelihood ratio $L(\boldsymbol{Y}^N) \stackrel{\text{def}}{=} \frac{P_1(\boldsymbol{Y}^N)}{P_0(\boldsymbol{Y}^N)}$ to a threshold and chooses H_1 only if it exceeds the threshold. The likelihood ratio for this problem can be computed as

$$\begin{aligned} L(\boldsymbol{Y}^N) &= \sum_{\boldsymbol{s}} P(\boldsymbol{s}) \frac{P_1(\boldsymbol{Y}^N|\boldsymbol{s})}{P_0(\boldsymbol{Y}^N)} \\ &= \sum_{\boldsymbol{s}} P(\boldsymbol{s}) \exp\Big(\beta \sum_{n=1}^{N} y_{s_n,n} - \frac{N\beta^2}{2}\Big), \end{aligned} \tag{3.1}$$

where $P(\boldsymbol{s}) = \pi_{s_1} p_{s_1,s_2} \cdots p_{s_{N-1},s_N}$ is the probability of the state sequence $\boldsymbol{s}$ under the Markov chain $\boldsymbol{P}$. Conditioned on the state sequence $\boldsymbol{s}$, the variable $y_{m,n}$'s distribution is different under the two hypotheses only if $m = s_n$. The expression in (3.1) might appear complicated, as the sum is over an exponentially large (M^N) number of possible state sequences. However, the likelihood ratio turns out to be

easy to compute: it can be reformulated in terms of matrix products[1]:

$$L(\boldsymbol{Y}^N) = \boldsymbol{\pi}^T \boldsymbol{D}_1 \boldsymbol{P} \boldsymbol{D}_2 \boldsymbol{P} \dots \boldsymbol{P} \boldsymbol{D}_N \mathbf{1}, \tag{3.2}$$

where $\boldsymbol{P}$ is the transition matrix of the Markov chain, and $\boldsymbol{D}_n$ is a diagonal matrix defined as

$$\boldsymbol{D}_n \stackrel{\text{def}}{=} \exp\left(-\tfrac{\beta^2}{2}\right) \operatorname{diag}\Big(\exp(\beta y_{1,n}), \dots, \exp(\beta y_{M,n})\Big)$$

for $1 \le n \le N$. Thus, the likelihood ratio can be computed in $\mathcal{O}(M^2 N)$ time.

A far more difficult problem is to characterize the performance of the detector, *i.e.*, to compute the type-I (false alarm) error probability $P_{\text{false\_alarm}}$ and the type-II (miss) error probability P_{miss}. Under the optimal detector, these are given by the expressions

$$P_{\text{false\_alarm}} = \iint \cdots \int_{L(\boldsymbol{Y}^N) > \tau} P_0(\boldsymbol{Y}^N)\, d^{MN}\boldsymbol{y}$$
$$P_{\text{miss}} = \iint \cdots \int_{L(\boldsymbol{Y}^N) < \tau} P_1(\boldsymbol{Y}^N)\, d^{MN}\boldsymbol{y}$$

where τ is the Neyman-Pearson threshold chosen to achieve the constraint on $P_{\text{false\_alarm}}$, and the integrals are over all MN variables $\{y_{m,n}\}$. These are very high dimensional integrals for which only Monte Carlo techniques would be practical. However, we would like to say something about the performance of these systems without having to simulate them. In particular, we expect that the performance

[1]Readers with a background in statistical physics may recognize this formula as an immediate consequence of the "transfer matrix" method [16] as applied to a one-dimensional generalized Potts model with a quenched random field. Those in the signal processing community may recognize the form of the conditional probability of a finite state discrete-time hidden Markov process.

depends on two parameters: the element-wise SNR β^2, and some measure of the complexity of the Markov chain $\boldsymbol{P}$. For example, more restrictive dynamics for the state sequence $\boldsymbol{s}$ should make it easier to correctly distinguish between the two hypotheses.

We consider the asymptotic performance of a detector as $N \to \infty$, *i.e.*, as the observation time increases without bound. Let $\epsilon \in (0, 1)$ be a constant. Given a sequence of optimal detectors $\delta_N(\boldsymbol{Y}^N)$ with false alarm constraint $P_{\text{false\_alarm}} \leq \epsilon$ (where δ_N has access to N observations of the network), the (type-II) *error exponent* is

$$\eta \stackrel{\text{def}}{=} -\lim_{N\to\infty} \frac{1}{N} \log P_{\text{miss}}(\delta_N). \tag{3.3}$$

This means that $P_{\text{miss}}(\delta_N) = \exp(-\eta N + o(N))$, so that the dominant feature of the miss probability is that it decays exponentially with a rate of η. In the remainder of this chapter, we will first prove that the error exponent in (3.3) is indeed a well-defined quantity, and then explore techniques to analytically characterize it.

3.2 The error exponent

3.2.1 Existence

The first question is whether the error exponent η is a well-defined quantity. If $\mathcal{H}_0$ and $\mathcal{H}_1$ were both i.i.d. hypotheses with single-letter marginal densities $p_0(\cdot)$ and $p_1(\cdot)$, then the Chernoff-Stein lemma [41] would tell us that $\eta = D(p_0||p_1) =$

$-\mathbb{E}_0 \log \frac{p_1(y)}{p_0(y)}$, the Kullback-Leibler divergence of p_1 from p_0. However, since $\mathcal{H}_1$ for our problem is not an i.i.d. hypothesis, the lemma in its original form is not applicable. So we prove the following generalization.

Lemma 3.1 (Generalized Chernoff-Stein Lemma)**.** *Suppose we have a sequence of hypotheses $\mathcal{H}_0^N$ and $\mathcal{H}_1^N$ with a well-defined Kullback-Leibler divergence rate*

$$\kappa \stackrel{def}{=} -\lim_{N\to\infty} \frac{1}{N}\mathbb{E}_0 \log \frac{P_1(\boldsymbol{Y}^N)}{P_0(\boldsymbol{Y}^N)} = -\lim_{N\to\infty} \frac{1}{N}\mathbb{E}_0 \log L(\boldsymbol{Y}^N).$$

Suppose furthermore that under $\mathcal{H}_0$, the normalized log likelihood ratio

$$\ell_N \stackrel{def}{=} \frac{1}{N}\log L(\boldsymbol{Y}^N)$$

converges in probability to the limit of its expectation, $-\kappa$. Then the error exponent η is well defined and $\eta = \kappa$.

Proof. See Appendix B.1. □

To apply Lemma 3.1 to our problem, we need to verify that its assumptions hold. This is established by the following proposition, which uses results from the theory of matrix-valued stochastic processes [54]:

Proposition 3.1. *The Kullback-Leibler divergence rate for our problem,*

$$\kappa = -\lim_{N\to\infty} \frac{1}{N}\mathbb{E}_0 \log \left(\boldsymbol{\pi}^T \boldsymbol{D}_1 \boldsymbol{P} \boldsymbol{D}_2 \boldsymbol{P} \dots \boldsymbol{P} \boldsymbol{D}_N \mathbf{1}\right),$$

exists. Further, under $\mathcal{H}_0$, the normalized log likelihood ratio converges almost surely:

$$\lim_{N\to\infty} \frac{1}{N}\log \left(\boldsymbol{\pi}^T \boldsymbol{D}_1 \boldsymbol{P} \boldsymbol{D}_2 \boldsymbol{P} \dots \boldsymbol{P} \boldsymbol{D}_N \mathbf{1}\right) \to -\kappa, \tag{3.4}$$

and thus it converges in probability.

Proof. We first note that, since $\boldsymbol{P}$ is a stochastic matrix, we have $\boldsymbol{P}\mathbf{1} = \mathbf{1}$, so we can add an extra factor of $\boldsymbol{P}$ into the expression (3.2) to obtain

$$L(\boldsymbol{Y}^N) = \boldsymbol{\pi}^T \boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{P} \boldsymbol{D}_N \boldsymbol{P} \mathbf{1}. \tag{3.5}$$

Under $\mathcal{H}_0$, the factors $\{\boldsymbol{D}_n \boldsymbol{P}\}_{n \geq 1}$ form an i.i.d. sequence of random matrices, with randomness induced by the Gaussian variables in the definition of $\boldsymbol{D}_n$. In a classical paper [54], Furstenberg and Kesten showed that for an i.i.d. sequence of random matrices $\boldsymbol{X}_n$, if $\mathbb{E} \log^+ \|\boldsymbol{X}_n\|_\infty$ is finite[2], the limit $\lim_{N\to\infty} \frac{1}{N} \mathbb{E} \log \|\boldsymbol{X}_1 \cdots \boldsymbol{X}_N\|_\infty$ exists and the random quantity $\frac{1}{N} \log \|\boldsymbol{X}_1 \cdots \boldsymbol{X}_N\|_\infty$ converges almost surely to the same limit. This quantity is equivalent to what is known as the (top) *Lyapunov exponent*—the exponential rate of growth or decay of a product of random matrices. First, let us show that the result applies to the factors $\{\boldsymbol{D}_n \boldsymbol{P}\}$. For any fixed n, we have:

$$\begin{aligned}
\mathbb{E} \log^+ \|\boldsymbol{D}_n \boldsymbol{P}\|_\infty &\leq \mathbb{E} \Big| \log \|\boldsymbol{D}_n \boldsymbol{P}\|_\infty \Big| \\
&= \mathbb{E} \Big| \log \max_m \Big\{ \exp(\beta y_{m,n} - \frac{\beta^2}{2}) \Big\} \Big| \\
&= \mathbb{E} \Big| \beta \max_m y_{m,n} - \frac{\beta^2}{2} \Big| < \infty.
\end{aligned}$$

So the condition we need to apply the Furstenberg-Kesten result holds. Now we must relate the likelihood ratio to the norm of the product of random matrices.

[2]Here, $\log^+(x) = \max\{0, \log(x)\}$, and the matrix ∞-norm is induced by the ℓ^∞ norm and is given by $\|\boldsymbol{X}\|_\infty \stackrel{\text{def}}{=} \max_i \sum_j |X_{i,j}|$.

Using Hölder's inequality, we have

$$\begin{aligned}
&\boldsymbol{\pi}^T \boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{P} \boldsymbol{D}_N \boldsymbol{P} \mathbf{1} \\
&\leq \|\boldsymbol{\pi}\|_1 \|\boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{P} \boldsymbol{D}_N \boldsymbol{P} \mathbf{1}\|_\infty \\
&= \|\boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{P} \boldsymbol{D}_N \boldsymbol{P}\|_\infty, \qquad (3.6)
\end{aligned}$$

where (3.6) follows from the definition of the matrix ∞-norm and the fact that all of the matrices involved are nonnegative and all of the vectors are positive. Meanwhile, as a lower bound, we let $\pi_{\min} = \min_m \pi_m$ and it holds that

$$\begin{aligned}
&\boldsymbol{\pi}^T \boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{P} \boldsymbol{D}_N \boldsymbol{P} \mathbf{1} \\
&\geq \pi_{\min} \|\boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{P} \boldsymbol{D}_N \boldsymbol{P} \mathbf{1}\|_1 \\
&\geq \pi_{\min} \|\boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{P} \boldsymbol{D}_N \boldsymbol{P}\|_\infty, \qquad (3.7)
\end{aligned}$$

where (3.7) again follows from the definition of the matrix ∞-norm. So we can sandwich the log likelihood ratio to within a vanishing constant:

$$\frac{1}{N} \log \|\boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{D}_N \boldsymbol{P}\|_\infty + \frac{1}{N} \log \pi_{\min} \leq \frac{1}{N} \log L(\boldsymbol{Y}^N) \leq \frac{1}{N} \log \|\boldsymbol{D}_1 \boldsymbol{P} \cdots \boldsymbol{D}_N \boldsymbol{P}\|_\infty.$$

The outer expressions converge almost surely and in expectation due to Furstenberg and Kesten's results [54, Theorems 1 and 2], so the log likelihood ratio must converge in the same way. □

Remark: Note that the proof only requires that the probability distribution of the initial state s_1 be positive at all nodes—there is no need to start from the stationary

distribution $\boldsymbol{\pi}$. In fact, since $\boldsymbol{P}$ is irreducible and aperiodic, we could relax the positivity constraint on the initial distribution and start with any known distribution.

Lemma 3.1 and Proposition 3.1 indicate that computing the error exponent boils down to computing the top Lyapunov exponent of products of random matrices, a problem known to be hard [126]. For $M \times M$ matrices, it generally requires solving an integral equation to obtain the invariant measure of a continuous diffusion process on a M-dimensional real projective space [35]. In low dimensions (*e.g.*, $M = 2$ or 3), this can be done with numerical quadrature (see, *e.g.*, [70, 81]), but this becomes intractable for high dimensional problems. Thanks to almost sure convergence of the normalized partial products in (3.4), one can use Monte Carlo simulations to estimate the error exponents. A simple Monte Carlo procedure that does just that is presented in Section 3.4.4, where we report some results of numerical simulations.

3.2.2 Upper and Lower Bounds

Obtaining analytical expressions for the error exponents for general Markov chain structures is expected to be a very challenging task. Instead, we will focus on deriving bounds for the error exponents. The Lyapunov exponent formulation of the error exponent as given in (3.4) does not lend itself to easy analysis. To proceed, we use the alternative form of the likelihood ratio in (3.1) to rewrite the

error exponent as follows

$$\begin{aligned}\eta &= \lim_{N\to\infty} -\frac{1}{N}\mathbb{E}\log\left(\sum_{\boldsymbol{s}} P(\boldsymbol{s})\exp\left(\beta y_{\boldsymbol{s}} - \frac{N\beta^2}{2}\right)\right) \\ &= \frac{\beta^2}{2} - \underbrace{\lim_{N\to\infty}\frac{1}{N}\mathbb{E}\log\Big(\sum_{\boldsymbol{s}} P(\boldsymbol{s})\exp(\beta y_{\boldsymbol{s}})\Big)}_{\varphi(\beta)},\end{aligned} \tag{3.8}$$

where $\boldsymbol{s} = (s_1, s_2, \ldots, s_N) \in \{1, \ldots, M\}^N$ is a state sequence of the Markov chain, and we define

$$y_{\boldsymbol{s}} \stackrel{\text{def}}{=} \sum_{n=1}^{N} y_{s_n,n} \sim \mathcal{N}(0, N) \tag{3.9}$$

to be the sum of the Gaussian random variables associated with a given state sequence $\boldsymbol{s}$. Here, and in what follows, we shall simply use $\mathbb{E}$ to refer to the expectation under $\mathcal{H}_0$, since we have no further use for $\mathbb{E}_1$. To study the behavior of the error exponent, we just need to study

$$\varphi(\beta) = \lim_{N\to\infty}\frac{1}{N}\mathbb{E}\log\Big(\sum_{\boldsymbol{s}} P(\boldsymbol{s})\exp(\beta y_{\boldsymbol{s}})\Big). \tag{3.10}$$

We will derive upper and lower bounds on this quantity, which will translate into bounds on the error exponent η. There is a simple lower bound: by treating the sum $\sum_{\boldsymbol{s}} P(\boldsymbol{s})\exp(\beta y_{\boldsymbol{s}})$ as an expectation and applying Jensen's inequality, we get

$$\varphi(\beta) \geq \lim_{N\to\infty}\frac{1}{N}\mathbb{E}\sum_{\boldsymbol{s}} P(\boldsymbol{s})\log\exp(\beta y_{\boldsymbol{s}}) = 0.$$

This then gives us an upper bound for the error exponent

$$\eta \leq \frac{\beta^2}{2},$$

which can also be interpreted as the "genie" bound: if we are given the true state sequence $\boldsymbol{s}$, then we can examine only the variables along that path and ignore all others, leading to an i.i.d. hypothesis testing problem with error exponent $\frac{\beta^2}{2}$. It provides an upper bound on the true error exponent since the extra side information about the correct path can only improve the performance.

To get a lower bound on η, we can still apply Jensen's inequality, but this time to the outer expectation $\mathbb{E}$ in (3.10), to obtain

$$\varphi(\beta) \leq \lim_{N\to\infty} \frac{1}{N} \log\Big(\sum_{\boldsymbol{s}} P(\boldsymbol{s})\mathbb{E}\exp(\beta y_{\boldsymbol{s}})\Big) = \frac{\beta^2}{2},$$

which gives us $\eta \geq 0$. Of course, this is trivial since η is equal to a limit of Kullback-Leibler divergences, which are always nonnegative. Another lower bound can be obtained by considering the test statistics $y_n = \sum_m y_{mn}$, the sums of the states in each time step. Since we are discarding information, the error exponent for this problem can be no greater than that for the original problem. But the new problem is just testing two i.i.d. hypotheses $y_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, M)$ and $y_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\beta, M)$. As we know, in the i.i.d. case the error exponent is simply the Kullback-Leibler divergence of these two densities, giving us a lower bound of

$$\eta \geq \frac{\beta^2}{2M}.$$

This is a nontrivial bound, but just barely. For large M, the error exponent is very small indeed. In fact, we would need M times the observation length to obtain the same performance as the genie-aided detector.

We will spend the remainder of this section and all of the next two sections computing a nontrivial lower bound for η, one that we will find empirically to fully capture its behavior. Qualitatively, this lower bound will guarantee that, above a certain threshold SNR, the error exponent will be bounded by

$$\eta \geq \frac{\beta^2}{2} - O(\beta),$$

meaning to leading order, the maximum likelihood detector will be just as good as the genie-aided detector.

To develop this bound, we will borrow ideas from the theory of spin glasses [87–89, 97, 122], a class of disordered systems studied in statistical physics. In fact, we have already chosen our notation so that our result closely resembles the quantities studied in that field. In particular, the function $\varphi(\beta)$ resembles the so-called "free energy density" of a spin glass, defined as

$$\phi(\beta) = -\frac{1}{\beta} \lim_{N\to\infty} \frac{1}{N} \mathbb{E} \log \Big(\sum_{\boldsymbol{s}} \exp(-\beta H(\boldsymbol{s})) \Big), \tag{3.11}$$

where N is the number of particles in the spin glass, $\boldsymbol{s} \in \mathbb{R}^N$ is an indexing vector representing the configurations of the system (there are typically exponentially large number of them), β is the inverse temperature parameter, and $H(\cdot)$ is a random Hamiltonian, a function defining the energy of each configuration. For our problem, we can write the function in (3.10) as $\varphi(\beta) = -\beta\phi(\beta)$ if we choose the Hamiltonian to be

$$H(\boldsymbol{s}) = -y_{\boldsymbol{s}} - \frac{1}{\beta} \log P(\boldsymbol{s}). \tag{3.12}$$

Despite the extra factor of $-\beta$, to be concise we will abuse the terminology and henceforth refer to $\varphi(\beta)$ for our problem as the free energy density.

Computing the free energy density of a disordered system is often very difficult. In fact, there are seemingly simple models that have been studied for many years with no exact solution [21, 122]. The main challenge lies in the fact that the free energy density $\phi(\beta)$ in (3.11) involves the sum of an exponentially large number of random variables. The high-dimensional *correlation structures* of the random Hamiltonians $\{H(\boldsymbol{s})\}_{\boldsymbol{s}}$ can often lead to remarkable phenomena (see, *e.g.*, [89, 99, 122]).

In our problem, the correlations of the Hamiltonians can be computed as follows. Let $\boldsymbol{s}^1, \boldsymbol{s}^2$ denote two arbitrary paths of the Markov chain, and let $H(\boldsymbol{s}^1), H(\boldsymbol{s}^2)$ be the associated Hamiltonians as defined in (3.12). Using (3.9), we can easily verify that

$$\operatorname{cov}(H(\boldsymbol{s}^1), H(\boldsymbol{s}^1)) = \mathbb{E}\, y_{\boldsymbol{s}^1} y_{\boldsymbol{s}^2} = \sum_{n=1}^{N} \mathbb{1}(s_n^1 = s_n^2), \tag{3.13}$$

where $\mathbb{1}(\cdot)$ is the indicator function. This means that the Hamiltonians of the various states in our problem are indeed correlated, and the covariance is equal to the number of times the two sequences overlap.

In the spin glass literature, removing or just reducing the correlations between state Hamiltonians can often simplify a problem [42, 122]. We follow this idea: if

we drop the correlations, we obtain a modified function[3]

$$\widetilde{\varphi}(\beta) = \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\log\Big(\sum_{\boldsymbol{s}} P(\boldsymbol{s})\exp(\beta x_{\boldsymbol{s}})\Big), \tag{3.14}$$

where $x_{\boldsymbol{s}} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, N)$, *i.e.* they are an uncorrelated Gaussian ensemble with the same variance as the $y_{\boldsymbol{s}}$. We note that the two functions in (3.14) and (3.10) have exactly the same form, the only difference being the absence of correlation in $\{x_{\boldsymbol{s}}\}$. Dropping the correlation, as we shall see, makes our problem tractable[4]. Interestingly, it also provides a lower bound on the error exponent, which is precisely what we seek for our problem. The argument relies on the following lemma:

Lemma 3.2 (Slepian's Lemma [122, pp. 12–15])**.** *Let the function $F : \mathbb{R}^L \to \mathbb{R}$ (for some L) satisfy the moderate growth condition*

$$\lim_{\|\boldsymbol{v}\|\to\infty} F(\boldsymbol{v})\exp(-a\|\boldsymbol{v}\|^2) = 0 \text{ for all } a > 0,$$

and have nonnegative mixed derivatives:

$$\frac{\partial^2 F}{\partial v_i \partial v_j} \geq 0 \text{ for } i \neq j.$$

Suppose that we have two independent zero-mean Gaussian random vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ taking values in $\mathbb{R}^L$ such that $\mathbb{E}x_i^2 = \mathbb{E}y_i^2$ and $\mathbb{E}y_iy_j \geq \mathbb{E}x_ix_j$ for $i \neq j$. Then $\mathbb{E}F(\boldsymbol{y}) \geq \mathbb{E}F(\boldsymbol{x})$.

[3]Strictly speaking, we need to show that $\widetilde{\varphi}(\beta)$ exists, *i.e.* that the limit is actually well-defined. We will do this in Section 3.4 by actually computing it. Until then, we presuppose its existence in all our arguments.

[4]In spin glass parlance, our function $\widetilde{\varphi}(\beta)$ may be regarded as the (rescaled) free energy density of a new generalization of the random energy model (REM) [42, 122].

Applying this to $\varphi(\beta)$ gives us the desired lower bound on the error exponent:

Proposition 3.2. *The error exponent satisfies* $\eta \geq \frac{\beta^2}{2} - \widetilde{\varphi}(\beta)$.

Proof. Define $F(\boldsymbol{v}) = -\log(\sum_{\boldsymbol{s}} P(\boldsymbol{s}) \exp(\beta v_{\boldsymbol{s}}))$. This is a function from $\mathbb{R}^{M^N}$ to $\mathbb{R}$ that clearly satisfies the moderate growth condition. We can compute the cross second derivative with respect to $v_{\boldsymbol{s}^1}$ and $v_{\boldsymbol{s}^2}$, with $\boldsymbol{s}^1 \neq \boldsymbol{s}^2$, as:

$$\frac{\partial F}{\partial v_{\boldsymbol{s}^1} \partial v_{\boldsymbol{s}^2}} = \frac{\beta^2 P(\boldsymbol{s}^1) P(\boldsymbol{s}^2) \exp(\beta(v_{\boldsymbol{s}^1} + v_{\boldsymbol{s}^2}))}{[\sum_{\boldsymbol{s}} P(\boldsymbol{s}) \exp(\beta v_{\boldsymbol{s}})]^2},$$

which is clearly nonnegative. From (3.13), we know that for $\boldsymbol{s}^1 \neq \boldsymbol{s}^2$, $\mathbb{E} y_{\boldsymbol{s}^1} y_{\boldsymbol{s}^2} \geq 0$, and we have constructed the $\boldsymbol{x}$ ensemble so that $\mathbb{E} x_{\boldsymbol{s}^1} x_{\boldsymbol{s}^2} = 0$. Thus, applying Slepian's Lemma gives us $\mathbb{E} F(\boldsymbol{y}) \geq \mathbb{E} F(\boldsymbol{x})$, which is equivalent to $\varphi(\beta) \leq \widetilde{\varphi}(\beta)$. The statement of the proposition then follows immediately from (3.8). □

Next, we will show how to explicitly compute $\widetilde{\varphi}(\beta)$ by using tools from large deviations theory. Before delving into the technical results, we first present in Section 3.3 a high-level and non-rigorous overview of the main ideas used in our approach. The discussions there also provide a roadmap to the various rigorous arguments that lead to our final results, stated as Theorem 3.3 and Propositions 3.6 and 3.7 in Section 3.4.

3.3 Main Ideas and Roadmap to the Technical Results

To begin, we can rewrite the free energy density as:

$$\widetilde{\varphi}(\beta) = \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\log \sum_{\boldsymbol{s}\in\mathcal{P}^N} \exp(\beta x_{\boldsymbol{s}} + \log P(\boldsymbol{s})), \tag{3.15}$$

where we are considering only the set $\mathcal{P}^N \subset \{1,\ldots,M\}^N$ of paths that have nonzero probability under the Markov chain $\boldsymbol{P}$ (the other paths contributed nothing to the sum in the first place.)

We can group the terms of the sum by their $\frac{1}{N}\log P(\boldsymbol{s})$ and $\frac{1}{N}x_{\boldsymbol{s}}$ values, dividing them into bins with a small width δ. Counting the number of configurations (*i.e.*, paths) in each bin as

$$C_N^\delta(\rho,\xi) \stackrel{\text{def}}{=} \#\{\boldsymbol{s}\in\mathcal{P}^N : \log P(\boldsymbol{s}) \in [N\rho, N(\rho+\delta)] \text{ and } x_{\boldsymbol{s}} \in [N\xi, N(\xi+\delta)]\},$$

then we should be able to approximate the sum as

$$\widetilde{\varphi}(\beta) \approx \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\log \sum_{\rho}\sum_{\xi} C_N^\delta(\rho,\xi)\exp(N[\beta\xi+\rho]), \tag{3.16}$$

where the sums are over cornerpoints of the bins. In Section 3.4, we will show that a form of this approximation can be made exact.

Of course, $C_N^\delta(\cdot,\cdot)$ is random due to its dependence on the Gaussian variables $\{x_{\boldsymbol{s}}\}$, but it turns out that there will be a concentration of measure phenomenon that will allow us to treat it deterministically in the large N limit. If we consider only the marginal count $C_N^\delta(\rho)$ of paths satisfying $\log P(\boldsymbol{s}) \in [N\rho, N(\rho+\delta)]$, then

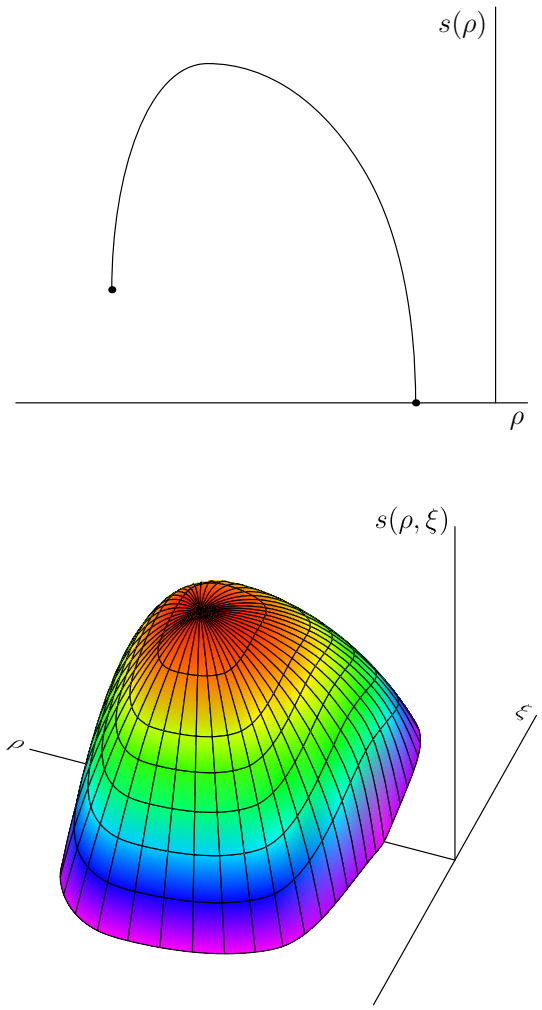


Figure 3.2: Notional illustrations of the microcanonical entropy densities $s(\rho)$ (top) and $s(\rho, \xi)$ (bottom). $s(\rho)$ is the exponential growth rate of the number of paths $\boldsymbol{s}$ satisfying $\frac{1}{N} \log P(\boldsymbol{s}) \approx \rho$, whereas $s(\rho, \xi)$ is, with probability 1, the exponential growth rate of the number of paths satisfying $\frac{1}{N} \log P(\boldsymbol{s}) \approx \rho$ and $\frac{1}{N} x_{\boldsymbol{s}} \approx \xi$. The density function $s(\rho, \xi)$ has a compact support, outside of which the density $s(\rho, \xi) = -\infty$, meaning that there is no path there. Analytical expressions for these functions are derived in Section 3.4.

there is no randomness involved; we can show that this count grows exponentially:

$$C_N^\delta(\rho) = \exp\Big(N[\sup_{\rho' \in [\rho, \rho+\delta]} s(\rho')] + o(N)\Big),$$

where $s(\rho)$ is the "microcanonical entropy density" function for $\frac{1}{N}\log P(\boldsymbol{s})$. This is physics jargon for the exponential growth rate of the number of configurations within an energy level [89]. In Section 3.4, we will show how to compute it (see Proposition 3.4) and derive several important properties (see Proposition 3.5). A notional illustration based on those properties is provided in Figure 3.2.

Meanwhile, the full count $C_N^\delta(\cdot,\cdot)$ will also grow exponentially:

$$C_N^\delta(\rho, \xi) = \exp\left(N\left[\sup_{\substack{\rho' \in [\rho, \rho+\delta] \\ \xi' \in [\xi, \xi+\delta]}} s(\rho', \xi')\right] + o(N)\right)$$

with probability 1 under the distribution of the $x_{\boldsymbol{s}}$, where $s(\rho, \xi)$ is the two-dimensional microcanonical entropy density function for the pair $(\frac{1}{N}\log P(\boldsymbol{s}), \frac{1}{N}x_{\boldsymbol{s}})$. In Section 3.4, we will show how to compute $s(\rho, \xi)$ (see Theorem 3.2), which is of course closely related to $s(\rho)$. Again a notional illustration is provided in Figure 3.2.

As N grows, the number of states grows exponentially, and we can let the bin width δ vanish and approximate the sum (3.16) by an integral. The free energy density can then be evaluated as

$$\begin{aligned}\widetilde{\varphi}(\beta) &\approx \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\log \iint \exp\left(N[s(\rho,\xi) + \beta\xi + \rho]\right) d\rho d\xi \\ &= \sup_{\rho,\xi}\Big\{s(\rho,\xi) + \beta\xi + \rho\Big\}, \end{aligned} \tag{3.17}$$

where the equality is obtained via the *Laplace principle*[5] [40]; we will use a rigorous formulation of this principle in Theorem 3.3 in the next section.

To actually compute $\widetilde{\varphi}(\beta)$, we will need to evaluate the supremum in (3.17). As it turns out, the microcanonical entropy density $s(\rho, \xi)$ has a compact support (see Figures 3.2 and 3.3), outside of which the density $s(\rho, \xi) = -\infty$. The supremum can thus be only achieved at the interior or the boundary of the support region. As illustrated in Figure 3.3, the location where the supremum is achieved depends on whether β is greater or less than a threshold of $\sqrt{2H}$, where H is the entropy rate of the Markov chain $\boldsymbol{P}$ (defined in Section 3.4.) As shown in the figure, below the threshold, the supremum is achieved at a critical point in the interior of the support region; as β increases the critical point moves up along the line $\rho = H$ until it hits the boundary. As β continues to increase beyond the threshold, the location of the supremum moves along the boundary in a direction of decreasing ρ. The change in behavior at the threshold corresponds to a *phase transition* in $\widetilde{\varphi}(\beta)$. In Section 3.4.3 we will provide a closed-form expression for $\widetilde{\varphi}(\beta)$ below the threshold, and a parametric representation for it above the threshold. The reader who wishes to skip the technical details can skip directly to that section, where we provide these expressions.

[5]The Laplace principle states that when N is very large, $\int \exp(Nf(x))dx = \exp(N \sup_x f(x) + o(N))$, *i.e.* the integral is dominated by the peak. This is also known as the *saddle-point technique*, a powerful tool in asymptotic integration [40].

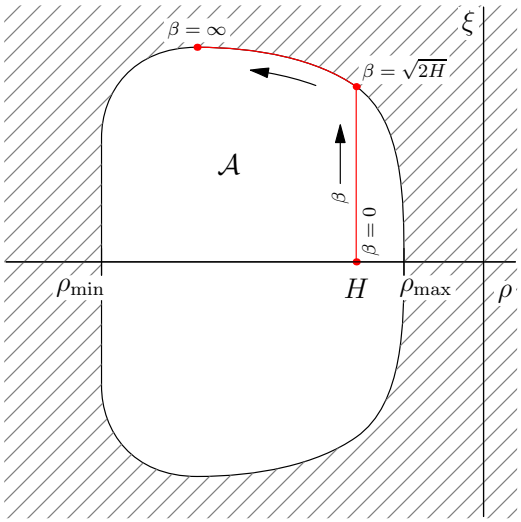


Figure 3.3: The location of the supremum that ultimately gives us $\widetilde{\varphi}(\beta)$ is illustrated here. The entropy density $s(\rho, \xi)$ is finite only in the compact region $\mathcal{A}$ illustrated here—this is also the effective domain of the large deviation rate function $I(\rho, \xi)$, which will be defined in (3.23). Below the threshold, the supremum in (3.17) is achieved at a critical point in the interior; above the threshold, the supremum moves along the boundary as β increases. The change in behavior at the threshold leads to a phase transition. Technical details will be provided in Section 3.4.

3.4 Rigorous Derivation

In this section, we use results from large deviations theory to rigorously derive expressions for the lower bound.

3.4.1 Large deviations and the microcanonical entropy density

First, we introduce the large deviations property for a sequence of probability measures:

Definition 3.1 (Large Deviation Property [48, pp. 35-36]). *Let $\mathcal{X}$ be a complete*

separable metric space and $\mathcal{B}(\mathcal{X})$ be the Borel σ-field of $\mathcal{X}$, Then the sequence $\{Q_N\}_{N=1}^{\infty}$ of probability measures on $\mathcal{B}(\mathcal{X})$ satisfies the large deviations property *if there is a lower semicontinuous function $I : \mathcal{X} \to [0, \infty]$ (the function may take the value ∞) with compact level sets such that*

1. $\limsup_{N\to\infty} \frac{1}{N} \log Q_N(B) \leq -\inf_{\boldsymbol{x}\in B} I(\boldsymbol{x})$ *for every closed set B in $\mathcal{B}$, and*
2. $\liminf_{N\to\infty} \frac{1}{N} \log Q_N(U) \geq -\inf_{\boldsymbol{x}\in U} I(\boldsymbol{x})$ *for every open set U in $\mathcal{B}$.*

$I(\boldsymbol{x})$ is known as the rate function.

To apply large deviations theory, we will consider the set of ordered pairs of the form $(\frac{1}{N}\log P(\boldsymbol{s}), \frac{1}{N}x_{\boldsymbol{s}})$ for $\boldsymbol{s} \in \mathcal{P}^N$ as inducing an empirical measure Q_N; for any set $B \subset \mathbb{R}^2$,

$$Q_N(B) \stackrel{\text{def}}{=} \frac{1}{\#\mathcal{P}^N} \# \left\{ \boldsymbol{s} \in \mathcal{P}^N : \left(\frac{1}{N}\log P(\boldsymbol{s}), \frac{1}{N} x_{\boldsymbol{s}} \right) \in B \right\}.$$

One way to think about this is as follows: if we choose an allowable state $\boldsymbol{s} \in \mathcal{P}^N$ uniformly at random (rather than choosing it by running the Markov chain), then $Q_N(B)$ is the probability that the ordered pair $(\frac{1}{N}P(\boldsymbol{s}), \frac{1}{N}x_{\boldsymbol{s}})$ is in B. This is just the number of states in the set B divided by the total number of allowable paths $\#\mathcal{P}^N$.

Since the $\{x_{\boldsymbol{s}}\}$ are random, Q_N itself is a *random* probability measure. It is important to note that there are two levels of randomness here: first, the random variables $\{\frac{1}{N}x_{\boldsymbol{s}}\}$ themselves, and second, the empirical probability distribution Q_N that they induce when paired with the log probabilities $\frac{1}{N}\log P(\boldsymbol{s})$. We will show

that *with probability 1*, the empirical probability measure will satisfy the large deviations property in Definition 3.1, and we will compute the rate function $I(\rho, \xi)$.

We will need to compute $\#\mathcal{P}^N$, the number of allowable paths. If every entry of transition matrix $\boldsymbol{P}$ is nonzero, then this is simple: $\#\mathcal{P}^N = M^N$. If each row of $\boldsymbol{P}$ has exactly K nonzero entries, meaning that each state can transition to only K other states, then $\#\mathcal{P}^N = K^N$. However, in the general case, we have:

$$\begin{aligned}\#\mathcal{P}^N &= \sum_{s_1}\sum_{s_2}\cdots\sum_{s_N} \mathbb{1}(p_{s_1,s_2} \neq 0)\mathbb{1}(p_{s_2,s_3} \neq 0)\cdots\mathbb{1}(p_{s_{N-1},s_N} \neq 0) \\ &= \mathbf{1}^T \left(\boldsymbol{P}^{(0)}\right)^{N-1} \mathbf{1},\end{aligned}$$

where for any matrix $\boldsymbol{A}$ and $t \in \mathbb{R}$, we define $\boldsymbol{A}^{(t)}$ to be the sparsity-preserving *Hadamard power* of $\boldsymbol{A}$, whose ijth entry is given by:

$$[\boldsymbol{A}^{(t)}]_{i,j} = \begin{cases} [\boldsymbol{A}]_{i,j}^t & \text{if } [\boldsymbol{A}]_{i,j} \neq 0 \\ 0 & \text{if } [\boldsymbol{A}]_{i,j} = 0. \end{cases}$$

In particular, $\boldsymbol{P}^{(0)}$ is a 0-1 matrix that is the adjacency matrix of the directed graph underlying the Markov chain. Its ijth element is 1 if and only if there is a nonzero probability of transitioning to state j directly from state i. Since $\boldsymbol{P}$ is irreducible and aperiodic, so must be $\boldsymbol{P}^{(0)}$. Due to the Perron-Frobenius theorem, $\lambda_{\max}(\boldsymbol{P}^{(0)})$ is simple, the associated left and right eigenvectors can be chosen to be positive, and all other eigenvalues are of smaller magnitude, so we can see that $\#\mathcal{P}^N$ grows exponentially with rate

$$\lim_{N\to\infty} \frac{1}{N} \log \#\mathcal{P}^N = \log \lambda_{\max}(\boldsymbol{P}^{(0)}).$$

The first step toward showing that Q_N satisfies the large deviation property with probability 1 is to show that its marginal Q_N^1 with respect to the first argument satisfies the large deviation property. This is simply the empirical probability measure on $\mathbb{R}$ induced by $\frac{1}{N}\log P(\boldsymbol{s})$ for all $\boldsymbol{s} \in \mathcal{P}^N$. It is *not* a random measure, since it does not depend on the Gaussian random variables $\{x_{\boldsymbol{s}}\}$. We will exploit the powerful Gärtner-Ellis theorem:

Theorem 3.1 (Gärtner-Ellis Theorem [48, p. 47])**.** *Suppose we have a sequence of random variables X_N taking values in $\mathbb{R}$. Let $\frac{1}{N}\log \mathbb{E}\exp(tX_N)$ be finite for every t, N. Suppose the limiting cumulant generating function (CGF), given by $c(t) \stackrel{def}{=} \lim_{N\to\infty}\frac{1}{N}\log \mathbb{E}\exp(tX_N)$, exists and is finite and differentiable for all t. Then the Legendre-Fenchel transform of $c(t)$, given by*

$$I(x) = \sup_{t\in\mathbb{R}} \Big\{ tx - c(t) \Big\},$$

is convex, lower semicontinuous, nonnegative, has compact level sets, satisfies $\inf_x I(x)$ = 0, and is the large deviations rate function for $\frac{1}{N}X_N$.

In our case, the random variable X_N is the one induced by choosing a state $\boldsymbol{s}$ uniformly at random from $\mathcal{P}^N$, and taking $X_N = \log P(\boldsymbol{s})$. We can compute the

limiting CGF as:

$$\begin{aligned} c(t) &= \lim_{N\to\infty} \frac{1}{N} \log \Bigg(\frac{1}{\mathbf{1}^T \left(\boldsymbol{P}^{(0)}\right)^{N-1} \mathbf{1}} \cdot \\ &\qquad \sum_{s_1} \cdots \sum_{s_N} \pi^t_{s_1} p^t_{s_1,s_2} \mathbb{1}(p_{s_1,s_2} \neq 0) \cdots p^t_{s_{N-1},s_N} \mathbb{1}(p_{s_{N-1},s_N} \neq 0) \Bigg) \\ &= \lim_{N\to\infty} \frac{1}{N} \log \left[\left(\boldsymbol{\pi}^{(t)}\right)^T \left(\boldsymbol{P}^{(t)}\right)^{N-1} \mathbf{1} \right] - \lim_{N\to\infty} \frac{1}{N} \log \left[\mathbf{1}^T \left(\boldsymbol{P}^{(0)}\right)^{N-1} \mathbf{1} \right] \\ &= \log \lambda_{\max}(\boldsymbol{P}^{(t)}) - \log \lambda_{\max}(\boldsymbol{P}^{(0)}), \end{aligned} \tag{3.18}$$

again using the Perron-Frobenius theorem, which due to the irreducibility and aperiodicity of $\boldsymbol{P}$ ensures that only the top eigenvalue remains for both terms. To apply the Gärtner-Ellis theorem, we need to show that $c(t)$ is differentiable. This follows from the following proposition, which provides several properties of the function $\log \lambda_{\max}(\boldsymbol{P}^{(t)})$ that we will need. To simplify the notation, we will define

$$\lambda_t \stackrel{\text{def}}{=} \lambda_{\max}(\boldsymbol{P}^{(t)}).$$

Proposition 3.3. *The function* $\log \lambda_t$ *satisfies the following properties:*

(1) $\log \lambda_t$ *is finite, analytic, and convex on* $\mathbb{R}$*.*

(2) $\log \lambda_t$ *is in fact* strictly *convex on* $\mathbb{R}$ *unless* $\boldsymbol{P}$ *is the transition matrix for a uniform random walk on a regular graph,* i.e. *there is some integer* $K \leq M$ *such that each row of* $\boldsymbol{P}$ *has exactly* K *nonzero entries, all of which are* $\frac{1}{K}$*. In that case,* $\log \lambda_t = (1-t)\log K$*.*

(3) Let $\boldsymbol{a}_t$ and $\boldsymbol{b}_t$ be the left and right Perron-Frobenius eigenvectors of $\boldsymbol{P}^{(t)}$, respectively. Then the derivative is given by:

$$\frac{d}{dt}\log\lambda_t = \frac{\boldsymbol{a}_t^T[(\log \boldsymbol{P}) \circ \boldsymbol{P}^{(t)}]\boldsymbol{b}_t}{\boldsymbol{a}_t^T[\boldsymbol{P}^{(t)}]\boldsymbol{b}_t}, \tag{3.19}$$

where the $\log$ *operates only on the nonzero entries of $\boldsymbol{P}$, and $\circ$ is the Hadamard (entrywise) product.*

(4) The range of $\frac{d}{dt}\log\lambda_t$ is given by

$$\inf_t \frac{d}{dt}\log\lambda_t = \liminf_{N\to\infty} \min_{\boldsymbol{s}\in\mathcal{P}^N} \frac{1}{N}\log P(\boldsymbol{s}) \stackrel{def}{=} \rho_{\min} \tag{3.20}$$

$$\sup_t \frac{d}{dt}\log\lambda_t = \limsup_{N\to\infty} \max_{\boldsymbol{s}\in\mathcal{P}^N} \frac{1}{N}\log P(\boldsymbol{s}) \stackrel{def}{=} \rho_{\max}. \tag{3.21}$$

Proof. See Appendix B.2. □

Now we can prove the following proposition:

Proposition 3.4. *Q_N^1 has a large deviations property with rate function*

$$\begin{aligned} I_1(\rho) &= \sup_t \{t\rho - \log\lambda_t + \log\lambda_0\} \\ &= \log\lambda_0 - s(\rho), \end{aligned}$$

where $s(\rho) \stackrel{def}{=} \inf_t \Big\{ \log\lambda_t - t\rho \Big\}$.

Proof. Since $\log\lambda_t$ is analytic, the limiting CGF $c(t)$ as defined in (3.18) is differentiable, and the proposition follows from the Gärtner-Ellis theorem. □

To complete the large deviations analysis, we will need to use several properties of $s(\rho)$. One quantity that will be important is the *entropy rate* of $\boldsymbol{P}$:

Definition 3.2. *The entropy rate of an irreducible and aperiodic Markov chain $\boldsymbol{P}$ is given by*

$$H = -\sum_i \pi_i \sum_j p_{i,j} \log p_{i,j},$$

where $\boldsymbol{\pi}$ is the unique stationary distribution. The entropy rate can be understood as the conditional entropy of the next state given the current state, averaged over the stationary distribution.

This definition will be important in the following proposition:

Proposition 3.5. *If $\boldsymbol{P}$ is the transition matrix for a uniform random walk on a K-regular graph, then $s(\rho)$ is given by*

$$s(\rho) = \begin{cases} \log K & \text{if } \rho = -\log K \\ -\infty & \text{if } \rho \neq -\log K. \end{cases} \tag{3.22}$$

Otherwise, $s(\rho)$ satisfies the following properties:

(1) $s : \mathbb{R} \to \mathbb{R} \bigcup \{-\infty\}$ is a concave function that is nonnegative on its effective domain, $[\rho_{\min}, \rho_{\max}]$, where $\rho_{\min}$ and $\rho_{\max}$ were defined in (3.20) and (3.21), respectively.

(2) $s(\rho)$ is continuous in $(\rho_{\min}, \rho_{\max})$, and continuous from above at $\rho_{\min}$ and $\rho_{\max}$.

(3) $s(\rho)$ is differentiable on $(\rho_{\min}, \rho_{\max})$. The function $s'(\rho)$ is one-to-one and $-s'(\rho)$ is the inverse function of $\frac{d}{dt} \log \lambda_t$.

(4) $s(-H) = H$ *and* $s'(-H) = -1$. *Meanwhile,* $s(\rho_0) = \log \lambda_0$ *and* $s'(\rho_0) = 0$*, where* $\rho_0 = \frac{(\boldsymbol{a}_0)^T (\log \boldsymbol{P}) \boldsymbol{b}_0}{\boldsymbol{a}_0^T \boldsymbol{b}_0}$.

Proof. See Appendix B.3. □

We provide notional illustrations of $\log \lambda_t$ and $s(\rho)$ in the general case, based on the properties described in Propositions 3.3 and 3.5, in Figure 3.4.

Now we can prove the large deviation property for the two-dimensional empirical measure Q_N induced by the pairs $(\frac{1}{N} \log P(\boldsymbol{s}), \frac{1}{N} x_{\boldsymbol{s}})$:

Theorem 3.2. *With probability* 1*, the empirical measure* Q_N *satisfies the large deviation property with rate function*

$$I(\rho, \xi) = \begin{cases} I_1(\rho) + \frac{\xi^2}{2}, & \text{if } I_1(\rho) + \frac{\xi^2}{2} \leq \log \lambda_0 \\ \infty, & \text{otherwise.} \end{cases} \tag{3.23}$$

Proof. See Appendix B.4. □

Remark: The microcanonical entropy density functions described in Section 3.3 and the large deviation rate functions computed in this section are closely related. Entropy density functions give the exponential growth rate for the number of states within some window; large deviation rate functions give the exponential decay rate for the probability of a uniformly chosen state in some window. Since the number of states in a window is equal to $\#\mathcal{P}^N$ times the probability under the empirical measure, we have that the microcanonical entropy density functions as illustrated

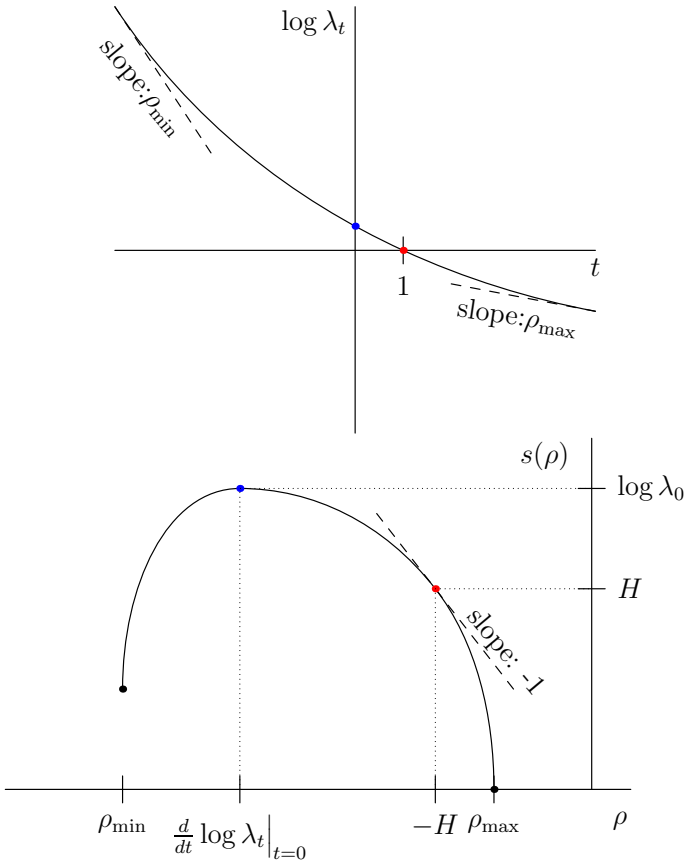


Figure 3.4: The basic properties of the functions $\log \lambda_t$ and $s(\rho)$ are illustrated here. $\log \lambda_t$ is a convex function (strictly convex except for a degenerate case); its value at $t = 1$ is 0, and it has limiting slopes $\rho_{\min}$ and $\rho_{\max}$. $s(\rho)$ is nonnegative and concave, takes the value H at $\rho = -H$ (where the slope is -1), and is finite only on $[\rho_{\min}, \rho_{\max}]$. Its peak and the location thereof is determined by the value and slope, respectively, of $\log \lambda_t$ at $t = 0$. (For the degenerate case of a uniform random walk on a K-regular graph, the curves look different: $\log \lambda_t$ is just a linear function $(1-t) \log K$, and $s(\rho)$ is only finite at a *single point*, $\rho = -\log K$, where $s(-\log K) = \log K$.)

in Figure 3.2 are given by:

$$s(\rho) = \log \lambda_0 - I(\rho)$$

and

$$\begin{aligned} s(\rho, \xi) &= \log \lambda_0 - I(\rho, \xi) \\ &= \begin{cases} s(\rho) - \frac{\xi^2}{2}, & \text{if } |\xi| \leq \sqrt{2s(\rho)} \\ -\infty, & \text{otherwise.} \end{cases} \end{aligned} \tag{3.24}$$

3.4.2 The saddle point technique through Varadhan's lemma

We can now compute the free energy density $\widetilde{\varphi}(\beta)$ given in (3.15). We rewrite it in terms of the empirical measure as:

$$\widetilde{\varphi}(\beta) = \lim_{N\to\infty} \frac{1}{N} \log(\#\mathcal{P}^N) + \lim_{N\to\infty} \frac{1}{N} \mathbb{E} \log \iint \exp(N[\beta\xi + \rho]) Q_N(d\rho, d\xi). \tag{3.25}$$

We have simply re-written the sum over all states as an integral over the discrete empirical measure induced by the states. The first term is, as we know, $\log \lambda_0$. The second term can be computed using Varadhan's lemma [48], a rigorous formulation of the Laplace principle (or the saddle point technique) applied to measures satisfying a large deviations property:

Lemma 3.3 (Varadhan's Lemma [48, p. 51]). *Suppose a sequence $\{Q_N\}_{N=1}^{\infty}$ of probability measures on $\mathcal{X}$ satisfies a large deviations property with rate function $I(\boldsymbol{x})$. Let $F : \mathcal{X} \to \mathbb{R}$ be a continuous function that satisfies the tail condition*

$$\lim_{L\to\infty} \limsup_{N\to\infty} \frac{1}{N} \log \int_{\boldsymbol{x}:F(\boldsymbol{x})\geq L} \exp(NF(\boldsymbol{x})) Q_N(d\boldsymbol{x}) = -\infty.$$

Then

$$\lim_{N\to\infty}\frac{1}{N}\log\int_{\mathcal{X}}\exp(NF(\boldsymbol{x}))Q_N(d\boldsymbol{x})=\sup_{\boldsymbol{x}\in\mathcal{X}}\Big\{F(\boldsymbol{x})-I(\boldsymbol{x})\Big\}.$$

We now have all the machinery in place to prove the main result:

Theorem 3.3. *The free energy density is given by*

$$\widetilde{\varphi}(\beta)=\sup_{\rho,\xi}\Big\{s(\rho,\xi)+\beta\xi+\rho\Big\}, \tag{3.26}$$

where $s(\rho,\xi)$ is the microcanonical entropy density given in (3.24).

Proof. To apply Varadhan's lemma, we need to show the tail condition

$$\lim_{L\to\infty}\limsup_{N\to\infty}\frac{1}{N}\log\iint_{(\rho,\xi):\beta\xi+\rho\geq L}\exp(N[\beta\xi+\rho])Q_N(d\rho,d\xi)=-\infty.$$

But this is simple. For all large enough L, the region $R=\{(\rho,\xi):\beta\xi+\rho\geq L\}$ has no intersection with the support of $I(\rho,\xi)$, and thus it satisfies $Q_N(R)=0$ with probability 1. Thus the tail condition holds, and Varadhan's lemma gives us that, almost surely,

$$\begin{aligned}\lim_{N\to\infty}\frac{1}{N}\log\sum_{\boldsymbol{s}\in\mathcal{P}^N}\exp(\beta x_{\boldsymbol{s}}+\log P(\boldsymbol{s}))&=\log\lambda_0+\sup_{\rho,\xi}\Big\{\beta\xi+\rho-I(\rho,\xi)\Big\}\\&=\sup_{\rho,\xi}\Big\{s(\rho,\xi)+\beta\xi+\rho\Big\}.\end{aligned} \tag{3.27}$$

In general, almost sure convergence does not guarantee the convergence of the expectation. However, if a sequence of random variables is *uniformly integrable*, then almost sure convergence (indeed, merely convergence in probability) guarantees convergence in L^1, which is stronger than convergence of the expectation.

Uniform integrability is a sort of joint tail condition for a sequence of random variables. As it turns out, the sequence of random variables $\frac{1}{N}\log\left(\sum_{\boldsymbol{s}} P(\boldsymbol{s})\exp(\beta x_{\boldsymbol{s}})\right)$ is uniformly integrable. Rather than belabor the point here, we will prove this fact (after formally defining uniform integrability) in Appendix B.5. This then immediately gives us the statement of the theorem. □

3.4.3 Evaluating the bound

Now we are in a position to actually compute $\widetilde{\varphi}(\beta)$, which will then give us a bound on the error exponent η. We start with the degenerate case, which has a closed form expression:

Proposition 3.6. *If $\boldsymbol{P}$ is the transition matrix for a uniform random walk on a K-regular graph, then the error exponent satisfies*

$$\eta \geq \begin{cases} 0, & \text{if } \beta \leq \sqrt{2\log K} \\ \frac{\beta^2}{2} - \beta\sqrt{2\log K} + \log K, & \text{otherwise}. \end{cases} \tag{3.28}$$

Proof. Combining (3.22), (3.24) and (3.26), we have $\widetilde{\varphi}(\beta) = \sup_{|\xi|\leq\sqrt{2\log K}}\left\{\beta\xi - \frac{\xi^2}{2}\right\}$. The supremum can be solved exactly; using the bound $\eta \geq \frac{\beta^2}{2} - \widetilde{\varphi}(\beta)$ gives us (3.28). □

The general case is slightly more complicated. We have the following parametric representation (of which the degenerate case expression given in Proposition 3.6 is

a special case):

Proposition 3.7. *For any irreducible and aperiodic Markov chain $\boldsymbol{P}$, the error exponent bound is*

$$\eta \geq \begin{cases} 0, & \text{if } \beta \leq \sqrt{2H} \\ \chi(\beta), & \text{if } \beta \geq \sqrt{2H}, \end{cases}$$

where $\chi(\beta)$ is a function that can be parametrized for $t \in (0, 1]$ as:

$$\begin{aligned} \beta_t &= \tfrac{\sqrt{2}}{t}\sqrt{\log \lambda_t - t\rho_t}, \\ \chi(\beta_t) &= \frac{1-2t}{t^2}\log \lambda_t - \frac{1-t}{t}\rho_t, \end{aligned} \tag{3.29}$$

and $\rho_t = \frac{d}{dt}\log \lambda_t$ is given in (3.19).

Proof. Since the function $s(\rho) - \frac{\xi^2}{2} + \beta\xi + \rho$ is concave and continuous on the effective domain of $s(\cdot,\cdot)$, given by $\mathcal{A} = \{(\rho, \xi) : |\xi| \leq \sqrt{2s(\rho)}\}$, the supremum is achieved at a point where $s'(\rho) = -1$ and $\xi = \beta$, if one exists in the interior of $\mathcal{A}$; if not, then the supremum is achieved on the boundary of $\mathcal{A}$. See Figure 3.3 for an illustration. From Proposition 3.5, we know that $s'(-H) = -1$ (the only such point), and $s(-H) = H$. So we get $\widetilde{\varphi}(\beta) = H - \frac{\beta^2}{2} + \beta^2 - H = \frac{\beta^2}{2}$ so long as $\beta \leq \sqrt{2H}$.

Otherwise, the supremum is achieved on the boundary, so $\xi = \sqrt{2s(\rho)}$ and

$$\widetilde{\varphi}(\beta) = \sup_{\rho \in [\rho_{\min}, \rho_{\max}]} \beta\sqrt{2s(\rho)} + \rho.$$

Since the function to be maximized is differentiable, the supremum occurs at the value of ρ for which

$$\frac{\beta s'(\rho)}{\sqrt{2s(\rho)}} + 1 = 0,$$

if one exists; otherwise the supremum occurs at one of the endpoints $\rho_{\min}$ or $\rho_{\max}$. We will show that such a point always exists. To see this, choose any $t \in (0, 1]$. Based on the results in Propositions 3.3 and 3.5, we know that for $\rho_t = \frac{d}{dt} \log \lambda_t$, we have $s'(\rho_t) = -t$ and $s(\rho_t) = \log \lambda_t - t\rho_t$. This in turn gives us a value of β:

$$\beta_t = -\frac{\sqrt{2s(\rho_t)}}{s'(\rho_t)}$$

and a corresponding value

$$\widetilde{\varphi}(\beta_t) = -\frac{2s(\rho_t)}{s'(\rho_t)} + \rho_t.$$

Using these representations, we can compute β_1—since we know that $\frac{d}{dt} \log \lambda_t \Big|_{t=1} = -H$, we have that $\beta_1 = \sqrt{2H}$. Meanwhile, $\lim_{t \to 0+} \beta_t = \infty$. This is because the numerator $\sqrt{2s(\rho_0)} = \sqrt{2 \log \lambda_0} > 0$ by the Perron-Frobenius theorem, so $s(\rho)$ is strictly positive in a neighborhood of $t = 0$, while the denominator $s'(\rho_t)$ approaches 0 from below. From the intermediate value theorem, we can then achieve any value of β in $[\sqrt{2H}, \infty)$ by choosing some $t \in (0, 1]$. Thus we have a fully parametric representation, and substituting the known values of $s(\rho_t)$ and $s'(\rho_t)$ and applying the bound $\eta \geq \frac{\beta^2}{2} - \widetilde{\varphi}(\beta)$ gives us the result. □

The bound given in Proposition 3.7 is equal to 0 when the SNR is below a threshold: $\beta^2 \leq 2H$. However, it is strictly positive for SNR above the threshold. Thus, we can guarantee strong performance when the SNR is greater than twice the entropy rate of the Markov chain. The entropy rate is smaller when the Markov structure is more restrictive; thus, the stronger our information about the dynamics of

the process, the stronger the performance of the detection. Furthermore, at very high SNR $\beta \gg 2H$, we can use the parametric representation (3.29) to show that $\frac{\beta^2}{2} - O(\beta) \leq \eta \leq \frac{\beta^2}{2}$, meaning the upper bound derived in Section 3.2.2 becomes tight. This is to be expected; at very high SNR, the knowledge of the true state path is not necessary to improve performance.

3.4.4 Numerical Verification

From Lemma 3.1, which equates the error exponent to the Kullback-Leibler divergence rate, and Proposition 3.1, which says the normalized log likelihood ratio converges almost surely to $-\kappa = -\eta$, and the fact that the log likelihood ratio can be computed efficiently, we have a simple Monte Carlo technique for estimating the true η. The only caveat is to prevent numerical underflow through a suitable renormalization procedure.

We used this Monte Carlo technique to estimate the error exponents over a range of SNRs for several Markov chains. In Figure 3.5 we compare the Monte Carlo simulations to the lower bound obtained using the parametric representation (3.29).

Although the phase transition appears only in the lower bound, the true error exponent curves appear to exhibit a smoothed version of the phase transition. Below the threshold the error exponent is quite small. It is bounded by the sum detector's error exponent of $\frac{\beta^2}{2M}$, as we showed in Section 3.2.2. Of course, the sum detector

Figure 3.5: Error exponent curves are plotted for random walks on four graphs, from top to bottom: a cycle graph with 101 vertices (H = 0.693 nats), a 32×32 grid (H = 1.58 nats), a random geometric graph with 1000 vertices (H = 2.09 nats), and a Watts-Strogatz small world graph [131] (H = 3.41 nats). The solid curve is the error exponent computed via Monte Carlo simulations. The green dashed curve is the sum-detector lower bound, which is barely nontrivial because M is large. The blue dashed curve is our statistical physics-based analytic lower bound, computed using the parametric representation (3.29). The analytic threshold (SNR = $2H$) is shown as well. At the same SNR level, the higher the entropy rate of the Markov chain, the worse the detector performance.

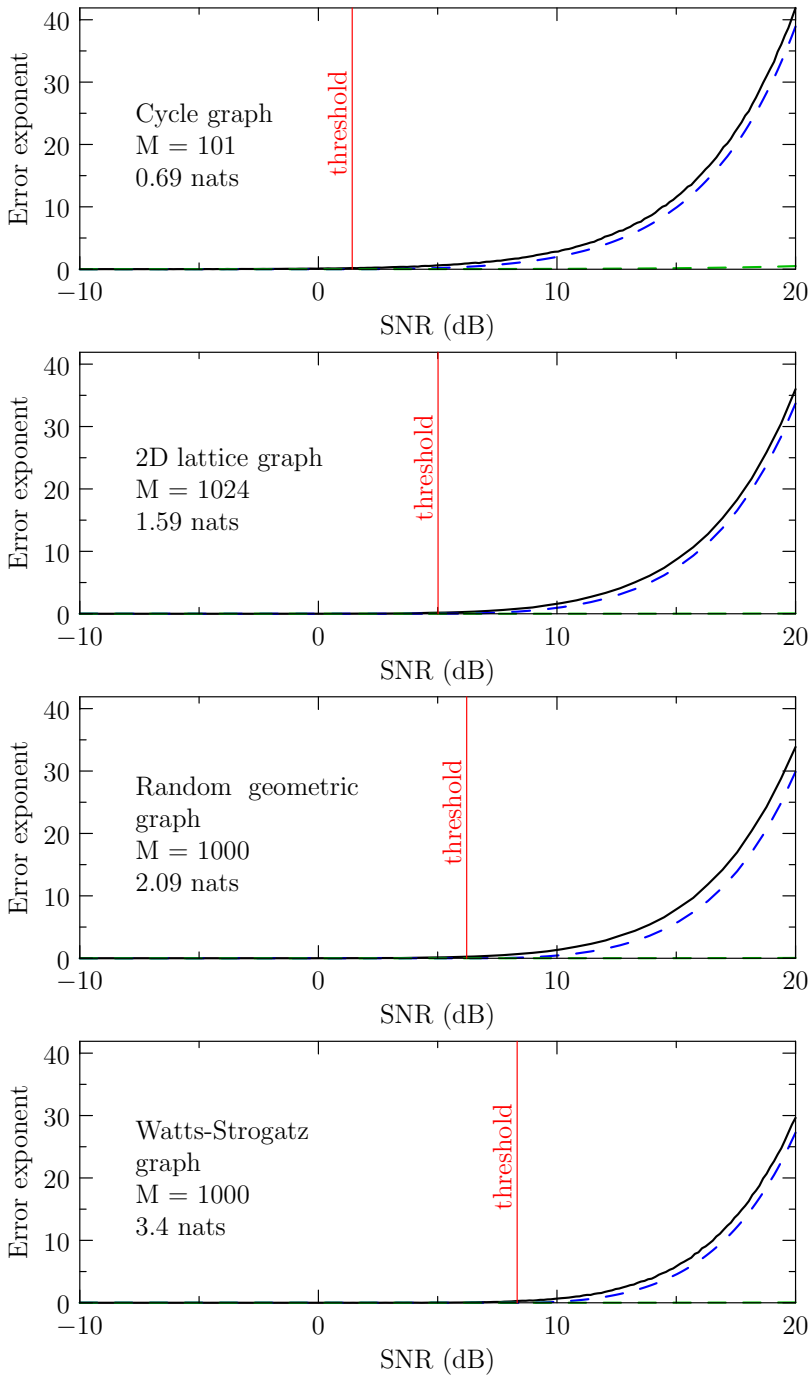


Figure 3.5 (Continued).

completely ignores the structure of the problem, and when M is large, this bound is practically 0. Meanwhile, above the threshold the error exponent grows quickly with increasing SNR. Thus the simple test $\beta^2 \lessgtr 2H$ suffices to determine whether one should expect good or bad detection performance.

3.5 Summary

In this chapter, we studied the problem of detecting a random walk on a graph from spatiotemporal measurements corrupted by Gaussian noise. We modeled the problem as a combinatorial hypothesis testing problem and studied the type-II error exponent of the optimal Neyman-Pearson detectors. We proved the existence of the error exponent and the fact that it is equal to the limiting Kullback-Leibler divergence rate between the two hypotheses. We showed how concepts from statistical physics could be used to analyze this quantity, and rigorously proved a bound for the error exponent. Monte Carlo simulations show that, unlike the sum detector bound, our bound fully captures the behavior of the error exponent. In particular, the bound provides us with a simple test for whether to expect strong or weak performance: if the SNR is greater than twice the entropy rate of the random walk, then detection will be easy.

Chapter 4

Exploring models for diffusion, epidemics, and influence

MANY DYNAMICAL SYSTEMS possess some kind of network structure. We have seen these kind of models before: in Chapter 2, we briefly studied a diffusion process inspired by the classical heat equation, and in Chapter 3, we considered a sequence of noisy measurements of a random walk on a graph. Other examples abound in the literature. There are models for epidemics spreading through human interaction or computer networks [129]. There are models for the spread of political ideas on social networks [13, 65]. There are models for developing distributed estimates through consensus on sensor networks [24]. In all of these models, the structure of the network controls the behavior of the process by requiring interactions to primarily occur between nodes that are connected in the network—the

dynamics are *local*.

In this chapter, we discuss two distinct problems related to dynamic processes on networks. In Section 4.1, we consider the problem of localization the source of an epidemic on a graph. In particular, we consider the susceptible-infected (SI) model [14], a well-known model in which each node is either infected with a disease or susceptible, and at each time step every susceptible node has some probability of being infected by an already-infected neighbor. In contrast to earlier work, we assume that we can only monitor a fraction of the nodes, but can observe them over a period of time. We develop a fast Monte Carlo technique to generate realizations from the model, and use it to perform inference on our observations.

Then, in Section 4.2, we introduce a logistic auto-regressive model (ALARM) for binary processes on networks. This is a very general model in which each node's next value depends probabilistically on the current value of its neighbors through a logistic link function. We show that this model can generate some very interesting behavior by showing phase transitions that depend on the underlying graph. We also show how to use regularized logistic regression to learn the underlying graph given a realization of this model. The model is general enough that it can approximate existing SI models, but simple enough to analyze more directly. We show that under a model mismatch, wherein the data is generated by an infection model but our inference is done assuming the ALARM model, we can still learn the underlying graph structure reasonably well.

4.1 Source Localization on Graphs

Epidemic models have long been studied by biologists and social scientists to study the spread of contagion on networks of varying scales and geometries [14]. These models range from the early, very simple models on (implicit) complete graphs [75], to very recent models on sophisticated random graphs [129]. This standard line of work has been mainly focused on the problem of determining the steady-state behavior of the model: will an epidemic die out, or will it remain active, with some fraction of the population always infected?

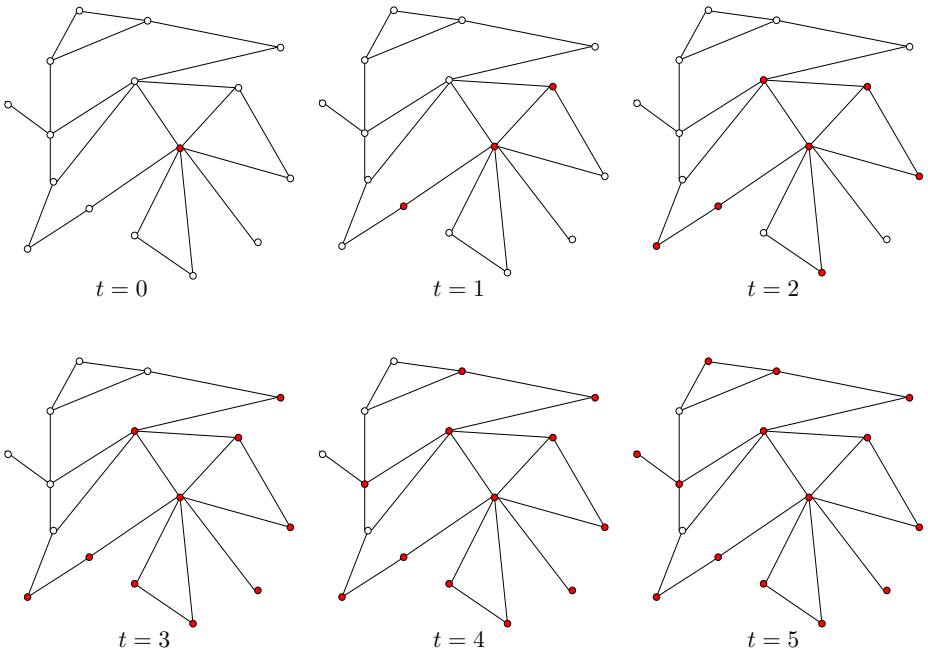


Figure 4.1: At time $t = 0$, a single vertex is infected (filled red circle) and the remaining ones are not. At each time step, an infected vertex has some probability of infecting each of its neighbors.

Recently, however, a new line of work has emerged to address the problem of detecting the source of an epidemic [52, 83, 84, 103, 104, 115, 116, 133]. In many ways, this is a much harder problem, which explains why it has only recently begun to receive attention. To determine the steady-state behavior, it is often enough to solve a system of differential equations. However, the source localization problem requires either high computational complexity to find near-optimal solutions, or simplified heuristics to achieve suboptimal performance. Often, algorithms are designed to work on trees, whose properties can greatly simplify the problem, and extended in an *ad hoc* manner to general graphs [83, 103, 116].

In this work, we describe an algorithm that operates on a sequence of observations made on a small subset of the nodes; our observation window begins at some unknown time after the initial infection. This algorithm is designed to function on general graphs, not just trees, and uses an initial Monte Carlo stage to estimate a pseudo-likelihood function for sources on a particular graph.

Serious investigation of the epidemic source localization problem began with Shah and Zaman [116] in 2011. They considered a continuous-time SI epidemic model on a graph, and sought to find the most likely source given the full set of infected nodes at some time. The infected nodes under this model form a connected subgraph, and they developed a metric called the *rumor centrality* for each node in the infection subgraph that served as a proxy for the likelihood of a particular node being the source. They provided an efficient message passing algorithm for

calculating the rumor centrality on a tree, and developed the breadth-first search (BFS) heuristic to approximate the rumor centrality on more general graphs. The epidemic source was chosen to be the member of the infected subgraph with the highest rumor centrality.

Others soon considered spectral techniques for source localization [52, 104]. Given the infected subgraph at some time, information about the likely source of infection can be gained from the eigendecomposition of the adjacency or Laplacian matrices of the subgraph. Luo and Tay [84] considered the case of multiple infection sources and developed an algorithm to detect these sources and determine the original infection source associated with each infected vertex. Seo *et al.* considered the case where only a subset of the vertices are monitored and developed four metrics most likely to be associated with the source; each successive metric breaks ties in the previous metrics [115]. Zhu and Ying [133] first considered the susceptible-infected-recovered (SIR) model in which nodes can recover from infection after some period of time and are then immune. They developed a message passing algorithm called the *reverse infection algorithm* that choose a source. More recently, Lokhov *et al.* have provided the statistical physics community's answer to the problem: a message passing algorithm on trees that computes the exact marginal probability for a vertex to be infected at some time given a source [83]. This facilitates the computation of a mean-field approximation to the likelihood of an observation. Though not exact on general graphs, it seems to give reasonable

results. Meanwhile, Pinto *et al.* considered the problem of locating the source given the exact time of infection for a set of observers. Their estimator is linear on trees, and is extended to general graphs using the BFS heuristic [103].

Our approach is the first to use multiple snapshots in a fixed, small time window for a sparse set of observers to estimate the source. Since the infection process we consider is Markov, if we could measure all the vertices, a snapshot of the state at a single time would be sufficient and any more data collection would be superfluous. However, because we only observe a small subset of the vertices, we can gain additional information by observing them over a period of time. Assuming that at least one observer sees a transition from susceptible to infected, our observation can be transformed into bounds on the difference between that observer's transition and all others. We introduce an alternate representation for the infection process in terms of the infection times for each vertex. The alternate representation allows us to quickly sample infection times conditioned on each source. This allows us to estimate the marginals of the observed relative infection times and compute a pseudolikelihood that is the product of those marginals, conditioned on each possible source.

4.1.1 The SI Epidemic Model

We start with a graph $G = (V, E)$, where V is the set of N vertices and E is the set of M edges, each of which is an unordered pair of vertices. If $\{v_i, v_j\} \in E$,

then v_i and v_j are in contact and capable of transmitting the infection to each other. We will use the notation $i \sim j$ to mean that v_i and v_j are neighbors on the graph. We consider a discrete-time version of the susceptible-infected (SI) model on G. At any time, each vertex is in one of two states, susceptible or infected. To model this mathematically, we will give every susceptible vertex the value 0 and every infected vertex the value 1. The epidemic model, illustrated in Figure 4.1, is a random process $\mathcal{X} = (\boldsymbol{x}(t))_{t=0}^{\infty}$ whose sample paths are sequences of binary vectors in $\{0,1\}^N$. We will assume that initially, a vertex $s \in V$ is chosen uniformly at random to be the infection source, and that the initial state vector $\boldsymbol{x}(0) = \boldsymbol{e}_s$ is 1 at s and 0 elsewhere.

At each time $t \in \{0, 1, \ldots\}$, any vertex that is infected remains so at time $t+1$, while any vertex that is susceptible becomes infected at time $t+1$ if it receives an infection signal from one of its neighbors on G. All infection signals are independent, and at each time step an infected vertex v_j sends an infection signal to susceptible neighbor v_i with probability λ_{ji}. Although the graph is undirected, we allow λ_{ji} and λ_{ij} to differ. The infection process is Markov; the vector $\boldsymbol{x}(t+1)$ depends on the previous states only through $\boldsymbol{x}(t)$. To simplify the notation, we will define $\mathcal{I}_i(t) = \{j : v_j \sim v_i, x_j(t) = 1\}$ to be the set of neighbors of i that are infected at

time t. Then the transition rule can be written as

$$\Pr\{x_i(t+1) = 1 | \boldsymbol{x}(t)\} = \begin{cases} 1 & \text{if } x_i(t) = 1 \\ 1 - \prod_{j \in \mathcal{I}_i(t)} (1 - \lambda_{ji}) & \text{if } x_i(t) = 0, \end{cases} \tag{4.1}$$

where the elements of $\boldsymbol{x}(t+1)$ are independent conditioned on $\boldsymbol{x}(t)$. $\mathcal{X}$ can be seen as a Markov chain with $2^N - 1$ states (if we ignore the all-susceptible state that is inaccessible from any other state); the distribution of $\boldsymbol{x}(t+1)$ conditioned on its history depends only on $\boldsymbol{x}(t)$, its state at time t. The all-infected state is an absorbing state, and will be reached in finite time with probability 1. The source localization problem we are interested in amounts to inferring the initial state of the Markov chain given partial measurements.

We can use an alternate representation of the process $\mathcal{X}$. Let $\tau_i = \min_t\{t : x_i(t) = 1\}$ be the time at which vertex i first becomes infected. Then $(\tau_1, \ldots, \tau_N)$ contains all of the information of $\mathcal{X}$, since we can reconstruct the sequence $(\boldsymbol{x}(0), \boldsymbol{x}(1), \ldots)$ from it. Using a randomly weighted version of the graph G, we can directly sample $(\tau_1, \ldots, \tau_N)$ given the source s.

Let H be the random, directed weighted graph $H = (V, E, W)$, where V and E are the vertex and edge sets for the graph G, and W consists of random weights w_{ij} and w_{ji} associated with each edge $\{i, j\}$ in E. H is directed because the weights for two different directions of an edge in E can be different. Each weight w_{ij} is a geometric random variable taking values in $\{1, 2, \ldots\}$ with parameter λ_{ij}. If v_i became infected at some time, and v_j were connected to no other vertices, then w_{ij}

represents the number of time steps it would take for v_j to become infected.

We define the following geodesic quasimetric on H:

$$d_H(v_i, v_j) = \min_{P \in \mathcal{P}_{ij}} \sum_{k=1}^{\text{len}(P)-1} w_{p_k, p_{k+1}}, \tag{4.2}$$

where $\mathcal{P}_{ij}$ is the set of paths on the graph from v_i to v_j, and

$$P = (p_1 = v_i, p_2, \ldots, p_{\ell(P)} = v_j)$$

. $d_H(v_i, v_j)$ is the length of the shortest possible path from v_i to v_j, where traversing an edge from v_k to v_l costs w_{kl}. It is a quasimetric and not a metric because it is not symmetric, due to the different weights on different directions of each edge. Now define $\mathcal{Y} = (\boldsymbol{y}(0), \boldsymbol{y}(1), \ldots)$ as follows:

$$s \sim \text{Unif}(1, \ldots, N); \quad y_i(t) = \begin{cases} 0 & \text{if } t < d_H(s, i) \\ 1 & \text{otherwise.} \end{cases} \tag{4.3}$$

$\mathcal{Y}$ is constructed to start with only the source s, chosen uniformly at random as in $\mathcal{X}$, infected, and then to have the infection time of each remaining vertex determined by its distance from the source on H. We have the following proposition:

Proposition 4.1. *The process $\mathcal{Y}$ is equal in distribution to $\mathcal{X}$,* i.e.

$$\Pr\{\boldsymbol{x}(0), \boldsymbol{x}(1), \ldots, \boldsymbol{x}(t)\} = \Pr\{\boldsymbol{y}(0), \boldsymbol{y}(1), \ldots, \boldsymbol{y}(t)\} \tag{4.4}$$

for every t.

Proof. First, it is clear that $\boldsymbol{y}(0) \overset{\text{d}}{=} \boldsymbol{x}(0)$ since each is a standard basis vector chosen uniformly at random. It remains to be shown that for each t, the distribution of

$\boldsymbol{y}(t+1)$ given the history $(\boldsymbol{y}(t), \ldots, \boldsymbol{y}(0))$ is identical to that for $\boldsymbol{x}(t+1)$ given $(\boldsymbol{x}(t), \ldots, \boldsymbol{x}(0))$. Along the way, we will show that $\mathcal{Y}$ is a Markov process.

First, we note that if $y_i(t) = 1$, then $\Pr\{y_i(t+1) = 1 | \boldsymbol{y}(t), \ldots, \boldsymbol{y}(0)\} = 1$, regardless of any of the other current or previous values in $\boldsymbol{y}$, since if $t \geq d_H(s,i)$, then $t+1 \geq d_H(s,i)$. On the other hand, if $y_i(t) = 0$, then

$$\Pr\{y_i(t+1) = 1 | \boldsymbol{y}(t), \ldots, \boldsymbol{y}(0), \text{with } y_i(t) = 0\} = 1- \tag{4.5}$$

$$\Pr\{d_H(s,i) > t+1 | d_H(s,i) > t, d_H(s,j) > t \, \forall y_j(t) = 0, d_H(s,j) = k_j \, \forall y_j(t) = 1\},$$

by the definition of $\mathcal{Y}$, where k_j is the time at which vertex j switched from 0 to 1 in the sequence $\boldsymbol{y}(0), \ldots, \boldsymbol{y}(t)$. From the definition of geodesic distance, we can rewrite $d_H(s,i) = \min_{j \sim i}(d_H(s,j) + w_{ji})$. Since $d_H(s,j) + w_{ji} > t+1$ automatically if $y_j(t) = 0$, we can consider in the minimum only those vertices that are infected at time t. So we get

$$\Pr\{y_i(t+1) = 1 | \boldsymbol{y}(t), \ldots, \boldsymbol{y}(0), \text{with } y_i(t) = 0\} \tag{4.6}$$

$$= 1 - \Pr\left\{ w_{ji} + d_H(s,j) > t+1 \, \forall j \in \mathcal{I}_i(t) \,\middle|\, \begin{array}{r} w_{ji} + d_H(s,j) > t \, \forall j \in \mathcal{I}_i(t), \\ d_H(s,j) = k_j \, \forall j \in \mathcal{I}_i(t) \end{array} \right\},$$

$$= 1 - \Pr\{w_{ji} > t+1-k_j \, \forall j \in \mathcal{I}_i(t) | w_{ji} > t - k_j, d_H(s,j) = k_j \, \forall j \in \mathcal{I}_i(t)\}. \tag{4.7}$$

Since the w_{ji} are geometric random variables, the conditional probability in (4.7) does not depend on the k_j's. Combined with the fact that the w_{ji}'s are independent,

we arrive at

$$\begin{aligned}&\Pr\{y_i(t+1)=1|\boldsymbol{y}(t),\ldots,\boldsymbol{y}(0),\text{with } y_i(t)=0\}\\&\quad=\Pr\{y_i(t+1)=1|\boldsymbol{y}(t)\}=1-\prod_{j\in\mathcal{I}_i(t)}(1-\lambda_{ji}),\end{aligned}$$

the same as the Markov transition rule for $\mathcal{X}$. Thus $\mathcal{Y}\stackrel{\mathrm{d}}{=}\mathcal{X}$. □

Thus, to sample $(\tau_1,\ldots,\tau_N)$ given a source s, we simply draw random weights w_{ij} and w_{ji} for each edge $\{i,j\}\in E$, and let $\tau_i=d_H(s,i)$. The distances can be computed using the Dijkstra shortest paths algorithm.

4.1.2 Problem statement

We now formally define the source localization problem. Suppose we can observe the states of L vertices $\mathcal{O}\subset V$ during an observation window $\{t_0,t_0{+}1,\ldots,t_0{+}T-1\}$. Without loss of generality, we assume $\mathcal{O}=\{1,2,\ldots,L\}$. We do not know t_0, so we have no knowledge of how long ago the infection began spreading from the source. Given the observations and knowledge of the graph and the infection parameters λ_{ji} we would like to estimate the source. If all vertices are observed, so that $L=N$, then $\boldsymbol{x}(t_0)$ is sufficient for estimation and we do not need to use the whole sequence of observations, since the infection sequence before t_0 is independent of the infection sequence after t_0 given the realization at t_0 thanks to the Markov property. This is no longer true if $L<N$ since in that case our observation sequence is a hidden Markov process (HMP) and does not itself satisfy the Markov

property [106].

Observations of a single vertex during our window may fall into three categories: we may observe its transition from susceptible to infected during our observation window, it may be susceptible during the entire window, or it may be infected during the entire window. This defines a partition of the set of observers $\mathcal{O} = \mathcal{O}_T \cup \mathcal{O}_S \cup \mathcal{O}_I$, where $\mathcal{O}_T$, $\mathcal{O}_S$, and $\mathcal{O}_I$ represent these categories, respectively. Suppose that $\mathcal{O}_T$ is nonempty, and without loss of generality, that $1 \in \mathcal{O}_T$. For each $i \in \mathcal{O}_T$, we let $m_i \in \{1, 2, \ldots, T-1\}$ be the index of the first observation for which the vertex transitions to the infected state (where our first observation has index 0). Because we do not have an absolute time reference, we have knowledge only about the relative infection times $\tau_i - \tau_1$ for $i \in \mathcal{O} \backslash \{i\}$.

Then we define a log-pseudolikelihood function ℓ by assuming that the relative delays of the observed nodes are independent:

$$\ell(s) = \sum_{i \in \mathcal{O}_T \backslash \{1\}} \log \Pr\{\tau_i - \tau_1 = m_i - m_1 | s\} + \sum_{i \in \mathcal{O}_I} \log \Pr\{\tau_i - \tau_1 \leq -m_1 | s\} + \sum_{i \in \mathcal{O}_S} \log \Pr\{\tau_i - \tau_1 \geq T - m_1 | s\}. \tag{4.8}$$

To estimate the marginals for the relative times, we will use Monte Carlo simulations to approximate the means and variances of the τ_i. We will use the procedure described in Section 4.1.1 to sample $(\tau_1, \ldots, \tau_L)$ for several iterations for each source s, then set $\mu_i(s)$ and $\sigma_i^2(s)$ to be the mean and variance of those samples. By approximating the infection times as Gaussian, the pseudo-likelihood function

(4.8) becomes

$$\ell(s) = -\frac{1}{2}\sum_{i\in\mathcal{O}_T\setminus\{1\}}\left(\log 2\pi\left(\sigma_i^2(s)+\sigma_1^2(s)\right)+\frac{(m_i-m_1-\mu_i(s)+\mu_1(s))^2}{\sigma_i^2(s)+\sigma_1^2(s)}\right) \quad (4.9)$$
$$+\sum_{i\in\mathcal{O}_I}\log\Phi\left(\frac{-m_1-\mu_i+\mu_1}{\sqrt{\sigma_i^2(s)+\sigma_1^2(s)}}\right)+\sum_{i\in\mathcal{O}_S}\log\left(1-\Phi\left(\frac{T-m_1-\mu_i+\mu_1}{\sqrt{\sigma_i^2(s)+\sigma_1^2(s)}}\right)\right),$$

where $\Phi(\cdot)$ is the cumulative distribution function for the standard normal distribution.

The computation is dominated by the Dijkstra shortest path algorithm in the sampling procedure for the infection times. By the central limit theorem, $O(1/\epsilon^2)$ samples are needed to achieve an error $o(\epsilon)$. The Dijkstra algorithm requires $O(M + N\log N)$ time to find the lengths of the shortest path to a single vertex from every other vertex. Thus, it takes $O(L(M + N\log N)/\epsilon^2)$ time to estimate $\mu_i(s)$ and $\sigma_i^2(s)$ for every $i \in \mathcal{O}$ and $s \in V$. In typical applications, graphs tend to be sparse, so we may take $M = o(N\log N)$, giving us a computational complexity of $O(LN\log(N)/\epsilon^2)$.

4.1.3 Numerical Results

To analyze the performance of this algorithm, we simulated the epidemic process on several graphs and computed the pseudolikelihoods for various observer fractions. The choice of performance metric is not entirely obvious: the probability of getting exactly the correct source is too pessimistic, since choosing, say, a neighbor of the true source is much better than choosing a source at the other end

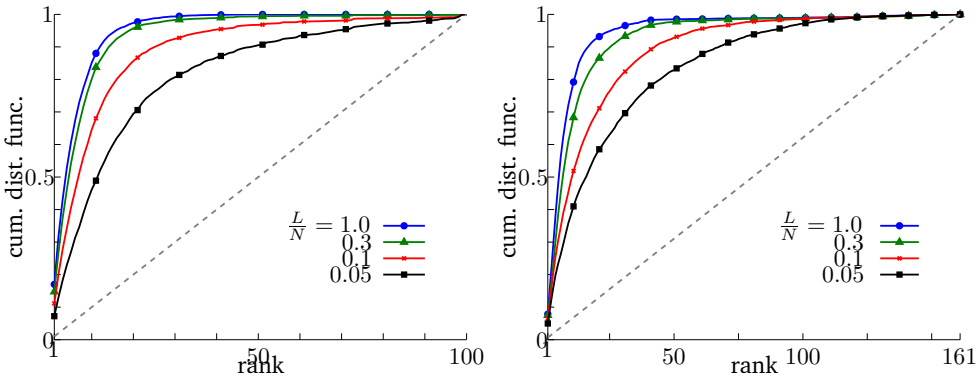


Figure 4.2: Numerical results from experiments involving two different graphs. On the left, a random geometric graph with 100 vertices; on the right, a 4-regular tree truncated 5 levels from a root vertex. For each graph, a series of trials was run in which a random source was chosen and a realization of $\mathcal{X}$ was generated. Our algorithm was given the realization from $t = 4$ to $t = 23$, for twenty observations, and computed pseudolikelihoods for each potential source. For the random geometric graph, the infection rate was 0.1 for every edge; for the tree it was 0.4. The CDF of the rank of the true source in the ordered list of pseudolikelihoods is shown here. Curves closer to the top right indicate better performance. On both graphs, the CDFs presented are for sampling factors of 1, 0.3, 0.1, and 0.05. It is evident that a lot of decimation is needed before the performance degrades appreciably. In all cases, the estimator does significantly better than the uniform random estimator, whose theoretical performance is given by the dotted line.

of the graph; yet average geodesic (or some other) distance is too optimistic, since many realistic graphs have small diameter. In prior work, most authors have instead considered the following metric: the vertices are sorted according to the value of whatever function is to be maximized, and the rank of the true source is considered.

In Figure 4.2, we show the cumulative distribution functions (CDFs) of the source's rank in our various experiments. We considered two graphs. First, a random geometric graph with 100 nodes randomly placed on the unit square and two

nodes connected if they are within a radius of 0.3 of one another; on this graph, an infection rate $\lambda = 0.1$ is used for every edge. Second, a 4-regular tree truncated after 5 levels; on this graph, an infection rate $\lambda = 0.4$ is used for every edge. In both cases, observations start at $t = 4$ and continue for 20 time steps. And in both cases, CDFs are computed when the fraction of observers is 1, 0.3, 0.1, and 0.05.

As expected, the performance of the estimator degrades as fewer vertices are observed. Interestingly, however, the degradation begins very slowly as the observer fraction drops from 1 down to 0.3, then speeds up as the fraction continues to fall. Note that on the 100-vertex random geometric graph, and observer fraction of 0.05 means that only 5 random vertices are observed. Meanwhile, it is interesting to see that for both graphs, observing only 30% of the vertices gives almost as good results as observing them all.

4.2 ALARM: A logistic auto-regressive model for binary processes on networks

The epidemic model described in Section 4.1 is just one kind of binary dynamic process on graphs. But these dynamic processes can be used to model systems in fields as varied as power systems engineering, political science, and ecology (see *e.g.* [13, 14]). In these systems, each node is in one of two states (which we will model as 0 and 1), and the current state of a node in the network is influenced by

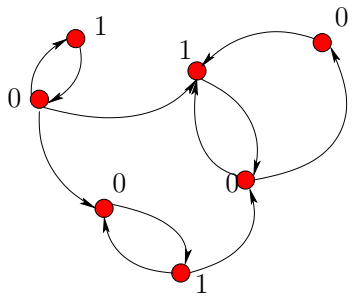


Figure 4.3: A snapshot of the ALARM model at time t. The model is defined on a directed graph which captures the local interactions of the nodes. Each node has a value of 0 or 1 at time t that depends on its neighbors' values at times $t-K,\ldots,t-1$.

the previous values of its neighboring nodes (and perhaps its own previous state). Several interesting questions arise in such models: we may wish to know whether they settle into some equilibrium, whether such an equilibrium is unique, whether the nodes are likely to coalesce to a single state, or even whether a small number of state flips can cascade across the network and transform the state of most of the nodes. As described in Section 4.1, recently some attention has turned to the problem of inferring the initial state. Furthermore, we may want to understand how well the network itself can be learned by merely observing the sequence of values produced by the model—a kind of system identification problem.

Over the years, several models for such dynamic processes have been developed in various fields (*e.g.*, [13, 59, 65]). In this chapter, we introduce a logistic autoregressive model (ALARM), a simple yet very flexible model for stochastic processes on graphs. The proposed ALARM model is a natural vector autoregressive process taking on binary values: at time t, the probability that a node has the value 1 is

the logistic function $\text{logit}^{-1}(\cdot) = \frac{\exp(\cdot)}{1+\exp(\cdot)}$ (a sigmoidal function illustrated in Figure 4.4) applied to a linear combination of its neighbors' (and its own) values at times $t - K, \ldots, t - 1$.

Like many existing models, it can model the influence of neighbors on a node's value. However, it is more general than existing models: it can capture negative influences (a node favors a value opposite a neighbor's), it can model uncertainty even when a node's neighbors are unanimous, it allows control over a node's bias toward one value or another, and it can model node values that depend strongly on their history.

This could be used to model the spread of a rumor on a social network with varying levels of skepticism or distrust, as an alternate model for the spread of an epidemic in a human interaction network, or the spread of a virus in a computer network. Its behavior can encapsulate that of existing models, but because it is more general, we can learn what kind of model best captures the behavior in a given system. Since each node's value is modeled by logistic regression against its neighbors' and its own preceding values, the model is open to analysis and existing logistic regression algorithms can be used for parameter estimation.

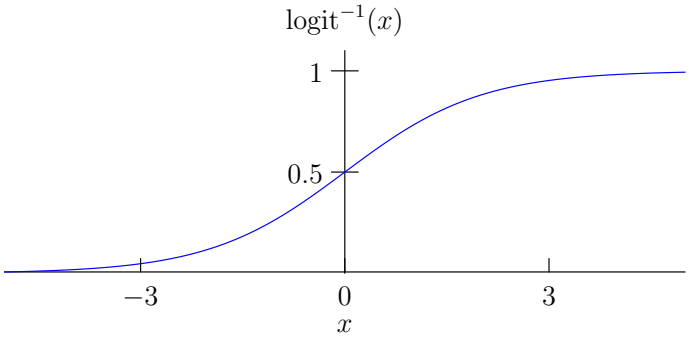


Figure 4.4: The logistic function is a sigmoidal function that maps the real line onto probabilities in (0,1). It is the "link function" in the ALARM model that takes a linear autoregressive expression and generates a probability of a Bernoulli variable being set to 1 in the next time step.

4.2.1 The ALARM Model

Definition

A logistic autoregressive model (ALARM) is defined on a directed graph $G = (V, E)$, where $V = \{v_1, \ldots, v_N\}$ is the set of N vertices and E is the set of directed edges, each of which is an ordered pair of vertices. We write $v_i \to v_j$ if $(v_i, v_j) \in E$, and $v_i \sim v_j$ if either $v_i \to v_j$ or $v_j \to v_i$. The indegree $\text{indeg}(v_i) = |\{v_j : v_j \to v_i\}|$ of a vertex is the number of incoming edges, the outdegree $\text{outdeg}(v_i) = |\{v_j : v_i \to v_j\}|$ is the number of outgoing edges, and the degree $\deg(v_i) = |\{v_j : v_i \sim v_j\}|$ is the total number of vertices connected to v_i one way or another. We define G^* as the undirected version of G, containing an edge $\{v_i, v_j\}$ if $v_i \sim v_j$ in G. We will assume that the indegree is bounded by a constant, so $\text{indeg}(v_i) < D$ for every i, and the $D \ll N$.

Under the ALARM model, we obtain a sequence of random vectors $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(T)} \in \{0,1\}^N$. Each element $y_i^{(t)}$ of $\boldsymbol{y}^{(t)}$ is independent of the others, conditioned on the previous K vectors $\boldsymbol{y}^{(t-1)}, \ldots, \boldsymbol{y}^{(t-K)}$, and takes the value 1 with probability

$$\begin{aligned} &\Pr\left(y_i^{(t)} = 1 \,\middle|\, \boldsymbol{y}^{(t-1)}, \ldots, \boldsymbol{y}^{(t-K)}\right) \\ &\qquad = \text{logit}^{-1}\left(\sum_{k=1}^{K}\sum_{j=1}^{N} h_{ij}^{(k)} y_j^{(t-k)} + b_i\right) \end{aligned} \tag{4.10}$$

where $\text{logit}^{-1}(x) = \frac{\exp(x)}{1+\exp(x)}$ is the logistic function [the inverse of the function $\text{logit}(x) = \log\left(\frac{x}{1-x}\right)$].

The parameters of the ALARM model are the K matrices $\boldsymbol{H}^{(1)}, \ldots, \boldsymbol{H}^{(K)}$ and the vector $\boldsymbol{b}$. Their elements are effectively logistic regression coefficients linking previous values of the dynamic process to the current values, giving the model its name: it is a vector autoregressive model with a logistic link function. This is the standard link function for generalized linear regression when the response variables are Bernoulli-distributed.

Our assumption is that the $\boldsymbol{H}^{(\cdot)}$ matrices respect the graph structure, *i.e.*, $h_{ij}^{(k)} \neq 0$ only if $v_j \to v_i$ or $i = j$. Of course, the model is well-defined even on a complete graph, which would allow for every coefficient to be nonzero. But on a true network, the model obeys the structure in a way that can be exploited, as we will show in Section 4.2.2. If we treat each time series $(y_i^{(1)}, \ldots, y_i^{(T)})$ as a random variable, the ALARM model is a graphical model described by the graph G^*, meaning that if $j \not\sim i$, then the time series at v_i is independent of the one at v_j conditioned on the time series at all of v_i's neighbors.

Properties

The ALARM process is a Kth order Markov chain with 2^N states. In general, such a Markov chain requires $2^{NK}(2^N - 1)$ real parameters to define. The ALARM model in general requires at most $N^2K + N$ real parameters (and only $NDK + N$ parameters under the bounded indegree condition). Despite its compact parametric representation, the ALARM model can capture a wide range of interactions.

Consider even just the special case of $K = 1$, so that the state at time t is dependent on the past only through the state at time $t-1$. If $h_{ij}^{(1)} > 0$, then $y_j^{(t-1)} = 1$ makes it more likely that $y_i^{(t)}$ will be 1. If $h_{ij}^{(1)} < 0$, then the opposite is true, and $y_i^{(t)}$ seeks the opposite state of $y_j^{(t-1)}$. If the diagonal element $h_{ii} > 0$, then y_i has "inertia" and may try to stay in the same state; if $h_{ii} < 0$, then y_i may oscillate between 1 and 0 (the specifics depend on the other coefficients and neighboring values).

The value of b_i is a kind of bias. If $b_i = 0$, then $y_i^{(t)} = 1$ with probability 1/2 if all of the neighbors $y_j^{(t-1)}$ were zero. $b_i > 0$ biases $y_i^{(t)}$ toward 1, and $b_i < 0$ biases it toward 0. Thus we can model behavior where neighbors influence each other either positively or negatively, nodes are biased one way or another, and nodes are either stuck in their current value or prone to flip-flopping. The meaning of the $\boldsymbol{H}$ and $\boldsymbol{b}$ parameters are illustrated in Figure 4.5.

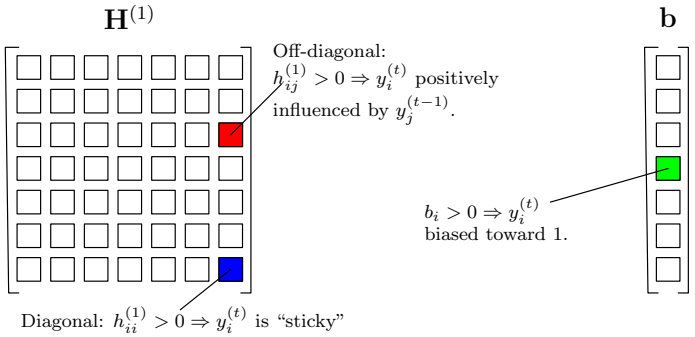


Figure 4.5: The meaning of the parameters of the ALARM model is illustrated here. The matrix $\boldsymbol{H}^{(1)}$ links the current states to the next states: if $h_{ij}^{(1)}$ is positive, then $y_i^{(t)}$ is positively influenced by $y_j^{(t-1)}$; if $h_{ii}^{(1)} > 0$, then y_i is "sticky", meaning that it resists changing its value; if $b_i > 0$, then $y_i^{(t)}$ is biased toward 1.

Examples

To illustrate some of the intriguing behavior that this model can produce, we consider the following special cases. Let $K = 1$, and suppose G is a 1D or 2D lattice graph (undirected) with N nodes. Let $\boldsymbol{A}$ be the adjacency matrix of the graph. For some $\beta > 0$ we define $\boldsymbol{H} = \beta(\boldsymbol{A} + \boldsymbol{I})$ and $\boldsymbol{b} = -\frac{1}{2}\boldsymbol{H}\mathbf{1}$. This value of $\boldsymbol{b}$ ensures the identity $\Pr\left(y_i^{(t)} = 1 \,\middle|\, \boldsymbol{y}^{(t-1)}\right) = \Pr\left(y_i^{(t)} = 0 \,\middle|\, \mathbf{1} - \boldsymbol{y}^{(t-1)}\right)$, so that flipping every state in $\boldsymbol{y}^{(t-1)}$ does the same to $\boldsymbol{y}^{(t)}$.

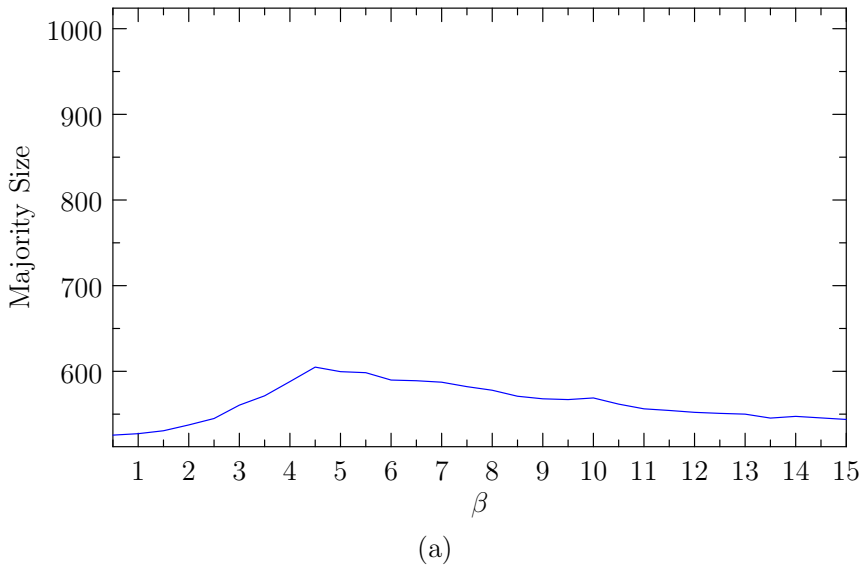
As in [13], we can use this to model influence in a social network. A node whose neighbors are evenly divided will have an equal chance of choosing either state. As the proportion of neighbors in a particular state deviates from that equilibrium, the logistic link function provides for an approximately linear response in the beginning; if the neighbors are nearly unanimous, the logistic function saturates and the

node is very likely to join them.

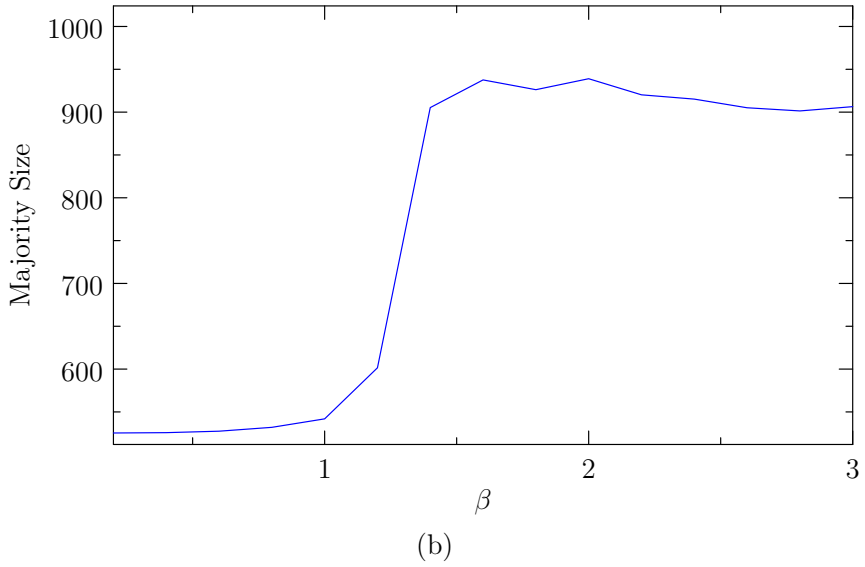
If the initial state $\boldsymbol{y}^{(0)}$ is i.i.d. Bernoulli$(1/2)$, then at time t, every state is as probable as its inverse. If we run the model for some time, does this mean that the final state will have an equal number of 0's and 1's? The question is a practical one: if we are modeling influence on a social network as in [13], then this tells us whether we settle into a consensus decision or a divided state. We might expect that for small β, the interactions are not strong enough to create a consensus, but as β increases, we end up with the vast majority of states either 0 or 1 (with each consensus equally probable).

We simulated the model to answer this question. The results for these two graphs are illustrated in Figure 4.6. We ran the model for 3000 time steps, and measured the size of the majority group. In each case, the graph size is 1024. Majority sizes near 512 indicate that no consensus is reached, whereas majority sizes nearer to 1024 indicate a consensus. A sharp phase transition is evident in the 2D lattice. As the interaction strength β increases past 1.3, we quickly move from a disordered phase to an ordered one with a strong consensus. Meanwhile, in the 1D case, even allowing β to go as high as 15 does not reveal any such phase transition. The final state is disordered even though the interaction strength is extremely strong.

This result hints at a connection to the Ising model of statistical physics [109]. A realization of the Ising model is a vector $\boldsymbol{z} \in \{-1, +1\}^N$ with probability given by



(a)



(b)

Figure 4.6: The size of majority after 3000 steps of the ALARM model is illustrated, for (a) 1D and (b) 2D lattice graphs. The initial states are i.i.d. Bernoulli$(1/2)$. The model parameters are $\boldsymbol{H} = \beta(\boldsymbol{A} + \boldsymbol{I})$, $\boldsymbol{b} = -\frac{1}{2}\boldsymbol{H}\mathbf{1}$, The 2D graph, unlike the 1D graph, has a phase transition. This is reminiscent of the behavior of the Ising model in physics [109].

$\Pr(\boldsymbol{z}) \propto \exp\left(\beta \boldsymbol{z}^T \boldsymbol{A} \boldsymbol{z}\right)$, where $\boldsymbol{A}$ is the adjacency matrix of the interaction graph of the system, and β is the inverse temperature. It is a well-known result in physics that the Ising model undergoes a similar phase transition to the one we observe in the ALARM model when the graph is a lattice of dimension 2 or greater, and that there is no phase transition on a 1D lattice [109]. The ALARM model is similar to Markov chain Monte Carlo techniques used to simulate the Ising model; but deeper study of the connection is warranted.

4.2.2 Parameter Estimation

In this section we present an algorithm for learning the parameters $\boldsymbol{H}^{(1)}, \ldots, \boldsymbol{H}^{(K)}$ and $\boldsymbol{b}$ of the ALARM model from a sequence of observations from the model. The log-likelihood of the ALARM model [conditioned on the initial states $\boldsymbol{y}^{(1-K)}, \ldots, \boldsymbol{y}^{(0)}$] is given by

$$\begin{aligned}
&\ell_{\{\boldsymbol{y}^{(t)}\}}\left(\boldsymbol{H}^{(1)}, \ldots, \boldsymbol{H}^{(K)}, \boldsymbol{b}\right) \\
&\quad = \sum_{t=1}^{T} \sum_{i=1}^{N} \left[y_i^{(t)} \left(\sum_{k=1}^{K} \sum_{j=1}^{N} h_{ij}^{(k)} y_j^{(t-k)} + b_i \right) \right. \\
&\qquad \left. - \log\left(1 + \exp\left(\sum_{k=1}^{K} \sum_{j=1}^{N} h_{ij}^{(k)} y_j^{(t-k)} + b_i \right)\right)\right] \qquad (4.11) \\
&\quad = \sum_{i=1}^{N} \ell^i_{\{\boldsymbol{y}^{(t)}\}}\left(h_{i\cdot}^{(1)}, \ldots, h_{i\cdot}^{(K)}, b_i\right), \qquad (4.12)
\end{aligned}$$

where the $\ell^i_{\{\boldsymbol{y}^{(t)}\}}$ are likelihoods for the parameters associated with the response of y_i to the neighboring values:

$$\begin{aligned}
&\ell^i_{\{\boldsymbol{y}^{(t)}\}}\left(h^{(1)}_{i1},\ldots,h^{(1)}_{iN},\ldots,h^{(K)}_{i1},\ldots,h^{(K)}_{iN},b_i\right)\\
&\quad\stackrel{\text{def}}{=}\sum_{t=1}^{T}\left[y_i^{(t)}\left(\sum_{k=1}^{K}\sum_{j=1}^{N}h_{ij}^{(k)}y_j^{(t-k)}+b_i\right)\right.\\
&\qquad\left.-\log\left(1+\exp\left(\sum_{k=1}^{K}\sum_{j=1}^{N}h_{ij}^{(k)}y_j^{(t-k)}+b_i\right)\right)\right].
\end{aligned}\tag{4.13}$$

The separability of the likelihood means we can learn the coefficients associated with the ith node independently of the others (but note that each independent log-likelihood uses all of the data.) This will simplify the analysis and allow for embarrassingly parallel algorithms to learn all the parameters. This learning really amounts to N logistic regression problems.

Let us consider the problem of learning the parameters associated with a single vertex: $h^{(1)}_{i\cdot},\ldots,h^{(K)}_{i\cdot}$ and b_i. The unknown graph structure described in Section 4.2.1 guarantees that for each k, the only non-zero variables out of $h^{(k)}_{i1},\ldots,h^{(k)}_{iN}$ are the D variables $h^{(k)}_{ij}$ for $j\to i$. This is a *group sparsity* [132] constraint on the parameter vector $\boldsymbol{\theta}=\left(\boldsymbol{\theta}_1^T,\ldots,\boldsymbol{\theta}_N^T\right)^T\stackrel{\text{def}}{=}\left(h^{(1)}_{i1},\ldots,h^{(K)}_{i1},\ldots,h^{(1)}_{iN},\ldots,h^{(K)}_{iN}\right)^T$. Unlike a sparsity constraint, which would limit the number of nonzero entries of $\boldsymbol{\theta}$, the group sparsity constraint limits the number of subvectors $\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_N$ that are not identically $\mathbf{0}$. Each subvector is associated with a neighboring vertex, and so at most D can be nonzero.

Directly incorporating this constraint into the maximum likelihood procedure

would result in a hard combinatorial problem. But we can use the standard approach of relaxing the constraint using the $\ell_{2,1}$ mixed norm defined by $||\boldsymbol{\theta}||_{2,1} = \sum_{i=1}^{N} ||\boldsymbol{\theta}_i||_2$ as a convex regularizer. The ℓ_2 part of the norm does not privilege any direction in the subspace associated with each vertex; but the ℓ_1 part of the norm promotes a group-sparse solution where only a small number of vertices are associated with non-zero values. We obtain the estimator

$$\left(\widehat{\boldsymbol{\theta}}, \widehat{b}_i\right) = \arg\min_{\boldsymbol{\theta}, b_i} \ell^i_{\{\boldsymbol{y}^{(t)}\}}(\boldsymbol{\theta}, b_i) + \lambda ||\boldsymbol{\theta}||_{2,1}, \tag{4.14}$$

or, more explicitly,

$$\begin{aligned} &\left(\widehat{h}^{(1)}_{i1}, \ldots, \widehat{h}^{(1)}_{iN}, \ldots, \widehat{h}^{(K)}_{i1}, \ldots, \widehat{h}^{(K)}_{iN}, \widehat{b}_i\right) \\ &\quad = \arg\min_{h^{(\cdot)}_{i\cdot}, b_i} \ell^i_{\{\boldsymbol{y}^{(t)}\}}(h^{(\cdot)}_{i\cdot}, b_i) + \lambda \sum_{j=1}^{N} \sqrt{\sum_{k=1}^{K} h^{(k)2}_{ij}}, \end{aligned} \tag{4.15}$$

where λ is a nonnegative regularization parameter. The regularization function does not include b_i because we have no reason to expect that $\boldsymbol{b}$ is sparse. The function to be minimized in (4.15) is convex, so it should be efficiently solvable. In fact, it is closely related to lasso and group-lasso logistic regression problems, for which several efficient algorithms exist [78, 86], and which can be shown to be consistent estimators [107].

To illustrate the utility of such techniques, we consider the problem of reconstructing the graph G from a realization of the ALARM model. The analogous problem for linear multivariate autoregressive models with Gaussian noise has been considered in [20]. Suppose we have a model with an unknown graph and $K = 1$.

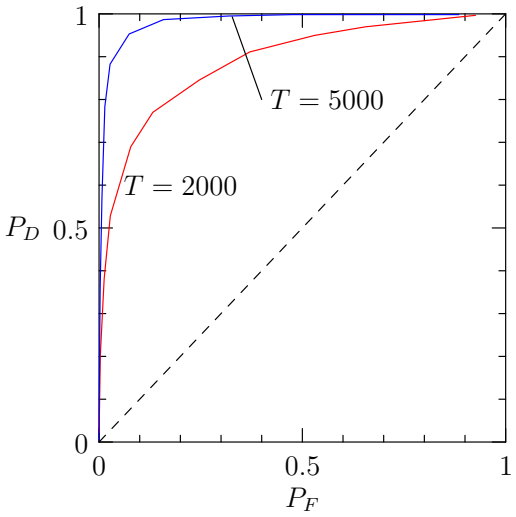


Figure 4.7: ROC curves for the detection of edges based on ALARM realizations are illustrated here. The realizations are generated with for a random geometric graph, with the strength of the nonzero edges (the values in the matrix $\boldsymbol{H}$) set to 0.2. The results based on 2000 and 5000 samples is shown.

If we use ℓ_1-regularized logistic regression to reconstruct each row of $\boldsymbol{H} \stackrel{\text{def}}{=} \boldsymbol{H}^{(1)}$, then we will obtain a matrix with many zero entries, due to the sparsity-recovery properties of the ℓ_1 regularization. As λ increases, more and more entries of $\widehat{\boldsymbol{H}}$ will be set to zero. If $v_j \to v_i$ but $\widehat{h}_{ij} = 0$, then we will characterize that as a misdetection; if $v_j \not\to v_i$ but $\widehat{h}_{ij} \neq 0$, then we will characterize it as a false alarm. Varying λ, we obtain a ROC curve. For the experiment, we used random geometric graph ($N = 100$, $D = 12$). We set $\boldsymbol{H} = \beta \boldsymbol{A}$, where $\boldsymbol{A}$ was the adjacency matrix of the graph, and $\beta = 0.2$. A realization of the ALARM model was generated with $T = 2000$ and $T = 5000$. We used a `l1_logreg`, a publicly available code for performing ℓ_1-regularized logistic regression [78]. The results are shown in Figure

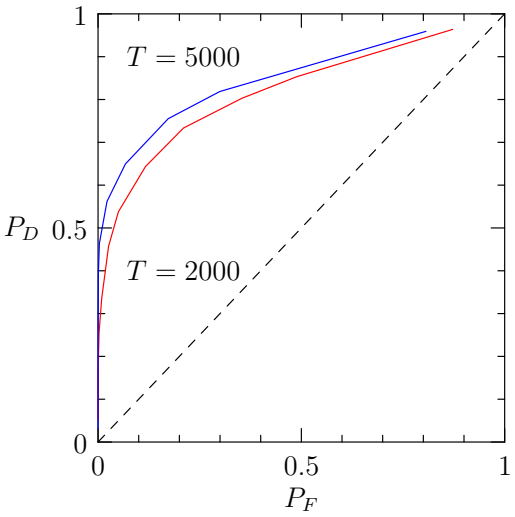


Figure 4.8: ROC curves for mismatched estimator when the data is generated by an SIS model. In both cases, the performance is far better than random chance. Estimation based on 5000 samples does not appear to be considerably better than estimation from only 2000 samples.

4.7.

The ALARM model is useful even when the underlying data is generated from a more specialized model. To show this, we conducted a separate experiment in which the data was generated from a susceptible-infected-susceptible (SIS) model. This model is similar to the SI model described in Section 4.1, except that each infected node has some positive probability of being "cured" and reverting to the susceptible state. This model is not a true specialization of the ALARM model, so ALARM cannot be used directly to model it. However, our results show that if the data is generated by an SIS model and we incorrectly assume it was generated by the ALARM model, we can still infer the graph structure. This will be useful in the

case that we know nothing about the dynamics of the process—ALARM is general enough that it can capture the qualitative behavior of the SIS model.

The performance of the ALARM graph inference technique under model mismatch is illustrated in Figure 4.8. We have plotted curves generated after 2000 samples and 5000 samples. The mismatched estimators perform far better than random chance.

4.3 Summary

In this chapter, we studied binary dynamic processes on networks, processes which are applicable to the study of many real-world phenomena. We first considered the problem of localizing the source of an epidemic on a graph. The epidemic was modeled using the standard SI model. We developed a Monte Carlo technique for quickly estimating the pseudolikelihood associated with each potential source node and used it to infer the true source. We then introduced ALARM, a simple but powerful model for binary processes on graphs in which a node's state is dependent on the states of its neighbors at preceding time steps through a logistic link function. We showed how to use ℓ_1-regularized logistic regression to estimate the graph structure from observations of the process. We also showed that the method works reasonably well under model mismatch, meaning it is able to infer the graph structure under the standard SI model.

Chapter 5

The Randomized Kaczmarz Algorithm: Exact MSE Analysis

THE KACZMARZ ALGORITHM [71], also known under the name Algebraic Reconstruction Technique (ART) [58], is a popular method for solving a large-scale overdetermined system of linear equations. Let

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}, \tag{5.1}$$

where $\boldsymbol{A}$ is a full-rank $m \times n$ matrix with $m \geq n$. Given $\boldsymbol{y} \in \mathbb{R}^m$, the algorithm proceeds to solve for $\boldsymbol{x}$ as follows: An initial guess $\boldsymbol{x}^{(0)}$ is chosen arbitrarily. The iterations then start with the first row, proceed in succession to the last row, and then cycle back to the first row, and so on. When row r is chosen, the current estimate $\boldsymbol{x}^{(k)}$ is projected onto the hyperplane $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}_r^T\boldsymbol{x} = y_r\}$ to obtain $\boldsymbol{x}^{(k+1)}$. Here, $\boldsymbol{a}_r^T$ is the rth row of $\boldsymbol{A}$.

Due to its simplicity, the Kaczmarz algorithm has been widely used in signal and image processing. It is also a special case of the projection onto convex sets (POCS) algorithm [125] for finding an intersection of many convex sets: in our case, we are looking for the intersection of a set of $(n-1)$-dimensional hyperplanes in $\mathbb{R}^n$.

It is well-known that the rate of convergence of the original Kaczmarz algorithm depends heavily on the exact ordering of the rows in $\boldsymbol{A}$ [64]. Recognizing this issue, Strohmer and Vershynin proposed in [120] a randomized Kaczmarz algorithm (RKA) that, instead of cycling sequentially through the rows in a deterministic fashion, chooses a row *at random* at each step. In their paper, they analyzed a specific probability distribution: choosing row i with probability proportional to its squared norm $||\boldsymbol{a}_i||^2$. They then showed the following upper bound on the mean squared error (MSE) of the RKA:

$$\mathbb{E}\|\boldsymbol{x}^{(N)} - \boldsymbol{x}\|^2 \leq \left(1 - \kappa_{\boldsymbol{A}}^{-2}\right)^N \|\boldsymbol{x}^{(0)} - \boldsymbol{x}\|^2, \tag{5.2}$$

where $\kappa_{\boldsymbol{A}} \stackrel{\text{def}}{=} \|\boldsymbol{A}\|_F \|\boldsymbol{A}^{-1}\|_2$ is the scaled condition number of $\boldsymbol{A}$, and $\boldsymbol{A}^{-1}$ is its left-inverse. Since $\kappa_{\boldsymbol{A}} \geq \sqrt{n}$, the above bound guarantees that the MSE decays *exponentially* as the RKA iterations proceed.

The work of Strohmer and Vershynin spurred a great deal of interest in RKA and its various extensions (see, *e.g.*, [30, 31, 38, 93, 95, 108, 134]). The original analysis in [120] assumes that the linear inverse problem is consistent (*i.e.*, noise-free). The noisy case was studied in [93]. A more general algorithm, involving random projections onto blocks of rows, was analyzed in [95]. Recently, Zouzias and Freris

[134] proposed a randomized extended Kaczmarz algorithm which converges to the least squares estimate of an inconsistent system of linear equations.

We provide three contributions in this chapter:

1. *An exact MSE formula:* All previous works on analyzing the performance of RKA provide strict upper bounds on the MSE. In this chapter, we present *exact* closed-form formulas for the MSE of RKA after k iterations, in both the noisy and noiseless case, for any k.

2. *Error floor:* We show that in the noise case, the MSE tends to a limiting value, an "error floor", and we show how to compute it.

3. *Annealed and quenched error exponents:* We provide an exact formula for the annealed error exponent, which measures the asymptotic rate of decay of the MSE in the noiseless case, and we provide a good approximation for the quenched error exponent, which measures the asymptotic rate of decay of the squared error during a typical realization of the algorithm.

4. *Optimal sampling probabilities:* Our exact MSE formula allows us to pose a simple semidefinite programming (SDP) problem, the solution of which leads to optimal row-selection probabilities to minimize the MSE of the RKA.

5.1 Performance Analysis

5.1.1 Overview of RKA

Given a matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and vector $\boldsymbol{y} \in \mathbb{R}^m$, the randomized Kaczmarz algorithm attempts to find a solution $\boldsymbol{x} \in \mathbb{R}^n$ to (5.1) as follows[1] The iterand $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$ is initialized arbitrarily. At each step k, a row r_k is chosen at random. The probability of choosing row i is p_i; the row-selection probabilities $p_1, \ldots, p_m$ are tunable parameters of the algorithm. The iterand is then updated according to the formula

$$\boldsymbol{x}^{(k)} = \boldsymbol{x}^{(k-1)} + \frac{y_{r_k} - \boldsymbol{a}_{r_k}^T \boldsymbol{x}^{(k-1)}}{||\boldsymbol{a}_{r_k}||^2} \boldsymbol{a}_{r_k}. \tag{5.3}$$

The algorithm is listed above. The intuition behind the algorithm is simple. Each row of $\boldsymbol{A}$ and its corresponding entry in $\boldsymbol{y}$ defines a hyperplane on which the solution $\boldsymbol{x}$ must lie; at each time step in the RKA algorithm we randomly select one of these hyperplanes and project the iterand onto it, getting closer to the true solution with each step.

5.1.2 Existing Bounds

Originally, Strohmer *et al.* proposed a specific probability distribution: $p_i = \frac{||\boldsymbol{a}_i||^2}{||\boldsymbol{A}||_F^2}$, where $||\cdot||_F$ is the Frobenius norm, and analyzed the behavior of the algorithm in terms of the properties of $\boldsymbol{A}$. However, the solution to (5.1) is invariant to

[1]The extension of the analysis in this chapter to the complex case is simple, but complicates the notation enough that we analyze only the real case here.

Randomized Kaczmarz Algorithm [120]

Require: $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ with rows $\boldsymbol{a}_1^T, \boldsymbol{a}_2^T, \ldots, \boldsymbol{a}_m^T$; $\boldsymbol{y} \in \mathbb{R}^m$; selection probabilities $p_1, \ldots, p_m$ with $\sum_i p_i = 1$; iteration count N.
Ensure: $\widehat{\boldsymbol{x}} \in \mathbb{R}^n$, an estimate for $\boldsymbol{x} \in \mathbb{R}^n$ solving $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$.
  Initialize $\boldsymbol{x}^{(0)}$ arbitrarily.
  for $k = 1$ to N **do**
    $r_k \leftarrow i$ with probability p_i.
    $\boldsymbol{x}^{(k)} \leftarrow \boldsymbol{x}^{(k-1)} + \frac{y_{r_k} - \boldsymbol{a}_{r_k}^T \boldsymbol{x}^{(k-1)}}{||\boldsymbol{a}_{r_k}||^2} \boldsymbol{a}_{r_k}$
  end for
  $\widehat{\boldsymbol{x}} \leftarrow \boldsymbol{x}^{(N)}$.

arbitrary and independent scalings of the rows. Thus, by looking at the properties of a rescaled version of $\boldsymbol{A}$, their analysis can be applied to arbitrary row-selection probabilities. Indeed, their results show that

$$(1 - 2N/\kappa_{\boldsymbol{A}}(p)^2) \leq \frac{\mathbb{E}\|\boldsymbol{x}^{(N)} - \boldsymbol{x}\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}\|^2} \leq (1 - \kappa_{\boldsymbol{A}}(p)^{-2})^N, \tag{5.4}$$

where $\kappa_{\boldsymbol{A}}(p) = \left\| \widetilde{\boldsymbol{A}}^{-1} \boldsymbol{D}_p^{-1/2} \right\|$, and we have defined $\widetilde{\boldsymbol{A}}$ as the row-normalized version of $\boldsymbol{A}$, and $\boldsymbol{D}_p$ as the diagonal matrix with $p_1, p_2, \ldots, p_m$ on the diagonal. $\widetilde{\boldsymbol{A}}^{-1}$ is the left-inverse, which is guaranteed to exist because $\boldsymbol{A}$ is a tall, full-rank matrix. This is sufficient to show that the error decays exponentially as the RKA iterations proceed. However, we will show that it is possible to compute the *exact* error after N iterations of RKA, for any $N \geq 1$, given the initial error. This will allow us to precisely characterize the rate of decay of the error.

The exponential decay of the error assumes that the measurements are exact, with no error. In general, however, the observations may be noisy, so that

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \tag{5.5}$$

where η is a noise vector. Needell extended the upper bound to the noisy case [94] and Zouzias and Freris improved it [135], obtaining

$$\mathbb{E}\|\boldsymbol{x}^{(N)} - \boldsymbol{x}\|^2 \leq (1 - \kappa_{\boldsymbol{A}}(p)^{-2})^N \|\boldsymbol{x}^{(0)} - \boldsymbol{x}\|^2 + \frac{\|\boldsymbol{\eta}\|^2}{\sigma_{\min}^2(\boldsymbol{A})}, \tag{5.6}$$

where $\sigma_{\min}(\boldsymbol{A})$ is the smallest singular value of $\boldsymbol{A}$. This is just the original noiseless bound plus an extra term proportional to the total squared error in the measurements, meaning that the error bound decays until it reaches a finite limiting value. Just as in the noiseless case, we will derive a formula for the exact MSE at each iteration. This will allow us to compute the exact "error floor", the actual limiting value of the MSE. Numerical results will show that in many cases, the bounds in both the noiseless and noisy cases are very loose, meaning that exact expressions can be valuable.

5.1.3 Exact MSE through the "lifting trick"

To lighten the notation in the sequel, we define the normalized ith row vector of $\boldsymbol{A}$ as $\widetilde{\boldsymbol{a}}_i \overset{\text{def}}{=} \frac{\boldsymbol{a}_i}{\|\boldsymbol{a}_i\|} \in \mathbb{R}^n$ and let $\widetilde{\eta}_i \overset{\text{def}}{=} \frac{\eta_i}{\|\boldsymbol{a}_i\|}$. Consider the noiseless case first. By combining (5.1) and (5.3), the error vector $\boldsymbol{z}^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{x}$ can be expressed as

$$\boldsymbol{z}^{(k)} = \boldsymbol{P}_{r_k} \boldsymbol{z}^{(k-1)}. \tag{5.7}$$

where $\boldsymbol{P}_i \overset{\text{def}}{=} \boldsymbol{I} - \widetilde{\boldsymbol{a}}_i \widetilde{\boldsymbol{a}}_i^T$ is the projection onto the $(n-1)$-dimensional subspace orthogonal to the ith row $\widetilde{\boldsymbol{a}}_i$. Averaging (5.7) over the randomness of the algorithm,

we get an iterative equation of the mean error vector

$$\mathbb{E}\boldsymbol{z}^{(k)} = \boldsymbol{P}\,\mathbb{E}\boldsymbol{z}^{(k-1)} \tag{5.8}$$

where $\boldsymbol{P} \overset{\text{def}}{=} \mathbb{E}\boldsymbol{P}_i$.

Note that the ease with which we can obtain (5.8) from (5.7) is mainly due to the *linearity* of the original random recursion in (5.7). However, the quantity we are interested in, the mean-squared error $\mathbb{E}\left\|\boldsymbol{z}^{(k)}\right\|^2$, is a non-linear (quadratic) term. To compute it, we "lift" the problem by treating the covariance matrix $\mathbb{E}\boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T$ as an n^2-dimensional vector whose dynamics are determined by the algorithm. In the lifted space, the dynamics are still linear (as we shall soon see), thus allowing for a relatively simple analysis. The MSE can be easily obtained as the trace of the matrix $\mathbb{E}\boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T$.

Consider the kth iteration:

$$\boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T = \boldsymbol{P}_i\boldsymbol{z}^{(k-1)}(\boldsymbol{z}^{(k-1)})^T\boldsymbol{P}_i \tag{5.9}$$

The linearity of this expression will be clearer if we "vectorize" $\boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T$ by vertically concatenating its columns to form a vector $\operatorname{vec}\left(\boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T\right) \in \mathbb{R}^{n^2}$. In what follows, we will make use of the following matrix identity which holds for any matrices dimensioned so that $\boldsymbol{ABC}$ is well-defined:

$$\operatorname{vec}(\boldsymbol{ABC}) = (\boldsymbol{C}^T \otimes \boldsymbol{A})\operatorname{vec}(\boldsymbol{B}), \tag{5.10}$$

where $\otimes$ represents the Kronecker matrix product [66]. First, we note that

$$\operatorname{vec}\left(\boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T\right) = \boldsymbol{z}^{(k)} \otimes \boldsymbol{z}^{(k)} \overset{\text{def}}{=} [\boldsymbol{z}^{(k)}]^{\otimes 2},$$

where $\boldsymbol{v}^{\otimes 2}$ is introduced as a shorthand notation for the Kronecker product of a vector $\boldsymbol{v}$ and itself. Then, we can apply the identity (5.10) to the right hand side of (5.9) to obtain

$$[\boldsymbol{z}^{(k)}]^{\otimes 2} = (\boldsymbol{P}_i \otimes \boldsymbol{P}_i)\,[\boldsymbol{z}^{(k-1)}]^{\otimes 2}$$

Taking expectation on both sides of the equation over the randomness of the algorithm, we obtain a simple iterative formula for the second-moment matrix:

$$\mathbb{E}[\boldsymbol{z}^{(k)}]^{\otimes 2} = \boldsymbol{Q}\,\mathbb{E}[\boldsymbol{z}^{(k-1)}]^{\otimes 2}. \tag{5.11}$$

where $\boldsymbol{Q} \overset{\text{def}}{=} \mathbb{E}\,(\boldsymbol{P}_i \otimes \boldsymbol{P}_i)$. So we can prove the following proposition:

Proposition 5.1. *After N iterations of the randomized Kaczmarz algorithm with initial iterand $\boldsymbol{x}^{(0)}$, the average error is given by*

$$\mathbb{E}\left|\left|\boldsymbol{x}^{(N)} - \boldsymbol{x}\right|\right|^2 = \operatorname{vec}(I_n)^T \boldsymbol{Q}^N \operatorname{vec}\left(\left(\boldsymbol{x}^{(0)} - \boldsymbol{x}\right)\left(\boldsymbol{x}^{(0)} - \boldsymbol{x}\right)^T\right).$$

Proof. We use the identity $\operatorname{tr}(\boldsymbol{A}^T\boldsymbol{B}) = \operatorname{vec}(\boldsymbol{A})^T \operatorname{vec}(\boldsymbol{B})$ to obtain

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 &= \mathbb{E}\operatorname{tr}\left(\boldsymbol{z}^{(N)}(\boldsymbol{z}^{(N)})^T\right) \\
&= \operatorname{vec}(\boldsymbol{I})^T \operatorname{vec}\left(\mathbb{E}\,\boldsymbol{z}^{(N)}(\boldsymbol{z}^{(N)})^T\right) \\
&= \operatorname{vec}(\boldsymbol{I})^T \boldsymbol{Q}^N \operatorname{vec}\left(\boldsymbol{z}^{(0)}(\boldsymbol{z}^{(0)})^T\right),
\end{aligned}$$

with the last step due to (5.11). This proves the proposition. □

Remark: $\boldsymbol{Q}$ is an $n^2 \times n^2$ matrix; however, due to its structure, it can be multiplied by a vector in $\mathbb{R}^{n^2}$ using $O(mn^2)$ operations rather than the naive $O(n^4)$.

Now consider the noisy case. Combining (5.3) and (5.5), we have this time that the error vector iteration is

$$\boldsymbol{z}^{(k)} = \boldsymbol{P}_{i_k} \boldsymbol{z}^{(k-1)} + \widetilde{\boldsymbol{a}}_{i_k} \widetilde{\eta}_{i_k}. \tag{5.12}$$

This gives us an iterative equation of the mean error vector

$$\mathbb{E}\, \boldsymbol{z}^{(k)} = \boldsymbol{P}\, \mathbb{E}\, \boldsymbol{z}^{(k-1)} + \boldsymbol{f}, \tag{5.13}$$

where we have defined $\boldsymbol{f} \overset{\text{def}}{=} \mathbb{E}\, \widetilde{\boldsymbol{a}}_i \widetilde{\eta}_i$. The dynamics of the error dyad $\boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T$, however, are more complicated. We have

$$\begin{aligned} \boldsymbol{z}^{(k)}(\boldsymbol{z}^{(k)})^T = \boldsymbol{P}_{i_k} \boldsymbol{z}^{(k-1)} (\boldsymbol{z}^{(k-1)})^T \boldsymbol{P}_{i_k} \\ + \widetilde{\eta}_{i_k} \left(\widetilde{\boldsymbol{a}}_{i_k} (\boldsymbol{z}^{(k-1)})^T \boldsymbol{P}_{i_k} + \boldsymbol{P}_{i_k} \boldsymbol{z}^{(k-1)} \widetilde{\boldsymbol{a}}_{i_k}^T \right) + \widetilde{\eta}_{i_k}^2 \widetilde{\boldsymbol{a}}_{i_k} \widetilde{\boldsymbol{a}}_{i_k}^T \end{aligned} \tag{5.14}$$

The error dyad couples with the error vector $\boldsymbol{z}^{(k)}$. If we use the identity (5.10) as before to rewrite this iteration, we obtain

$$[\boldsymbol{z}^{(k)}]^{\otimes 2} = (\boldsymbol{P}_{i_k} \otimes \boldsymbol{P}_{i_k})\, [\boldsymbol{z}^{(k-1)}]^{\otimes 2} + \widetilde{\eta}_{i_k} \left(\boldsymbol{P}_{i_k} \otimes \widetilde{\boldsymbol{a}}_{i_k} + \widetilde{\boldsymbol{a}}_{i_k} \otimes \boldsymbol{P}_{i_k} \right) \boldsymbol{z}^{(k-1)} + \widetilde{\eta}_{i_k}^2 \widetilde{\boldsymbol{a}}_{i_k}^{\otimes 2}.$$

Taking expectation on both sides of the equation over the randomness of the algorithm, we obtain a simple iterative formula for the second-moment matrix:

$$\mathbb{E}[\boldsymbol{z}^{(k)}]^{\otimes 2} = \boldsymbol{Q}\, \mathbb{E}[\boldsymbol{z}^{(k-1)}]^{\otimes 2} + \boldsymbol{D}\, \mathbb{E}\, \boldsymbol{z}^{(k-1)} + \boldsymbol{e}, \tag{5.15}$$

where we introduce $\boldsymbol{D} \overset{\text{def}}{=} \mathbb{E} \widetilde{\eta}_i \left(\boldsymbol{P}_i \otimes \widetilde{\boldsymbol{a}}_i + \widetilde{\boldsymbol{a}}_i \otimes \boldsymbol{P}_i \right)$ and $\boldsymbol{e} \overset{\text{def}}{=} \mathbb{E} \widetilde{\eta}_i^2\, \widetilde{\boldsymbol{a}}_i^{\otimes 2}$.

We can combine (5.13) and (5.15) into a single linear recursion over the state variable $\left[([\boldsymbol{z}^{(k)}]^{\otimes 2})^T,\ (\boldsymbol{z}^{(k)})^T\right]^T$:

$$\begin{pmatrix} \mathbb{E}[\boldsymbol{z}^{(k)}]^{\otimes 2} \\ \mathbb{E}\boldsymbol{z}^{(k)} \end{pmatrix} = \boldsymbol{H} \begin{pmatrix} \mathbb{E}[\boldsymbol{z}^{(k-1)}]^{\otimes 2} \\ \mathbb{E}\boldsymbol{z}^{(k-1)} \end{pmatrix} + \begin{pmatrix} \boldsymbol{e} \\ \boldsymbol{f} \end{pmatrix}, \tag{5.16}$$

where

$$\boldsymbol{H} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{Q} & \boldsymbol{D} \\ \boldsymbol{0} & \boldsymbol{P} \end{pmatrix}. \tag{5.17}$$

We thus have the following proposition:

Proposition 5.2. *For a fixed noise vector $\boldsymbol{\eta}$, and an initial error vector $\boldsymbol{z}^{(0)}$, the MSE of RKA at the Nth iteration is given by*

$$\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 = \begin{pmatrix} \operatorname{vec}(\boldsymbol{I}_n) \\ \boldsymbol{0}_n \end{pmatrix}^T \left[\boldsymbol{H}^N \begin{pmatrix} [\boldsymbol{z}^{(0)}]^{\otimes 2} - \boldsymbol{v}_1 \\ \boldsymbol{z}^{(0)} - \boldsymbol{v}_2 \end{pmatrix} + \begin{pmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{pmatrix} \right], \tag{5.18}$$

where $\boldsymbol{v}_1 = (\boldsymbol{I} - \boldsymbol{Q})^{-1}[\boldsymbol{e} + \boldsymbol{D}\boldsymbol{v}_2]$ and $\boldsymbol{v}_2 = (\boldsymbol{I} - \boldsymbol{P})^{-1}\boldsymbol{f}$.

Proof. We first solve the linear recursion (5.16) to get a closed-form expression

$$\begin{pmatrix} \mathbb{E}[\boldsymbol{z}^{(N)}]^{\otimes 2} \\ \mathbb{E}\boldsymbol{z}^{(N)} \end{pmatrix} = \boldsymbol{H}^N \begin{pmatrix} \mathbb{E}[\boldsymbol{z}^{(0)}]^{\otimes 2} \\ \mathbb{E}\boldsymbol{z}^{(0)} \end{pmatrix} + \sum_{k=0}^{N-1} \boldsymbol{H}^k \begin{pmatrix} \boldsymbol{e} \\ \boldsymbol{f} \end{pmatrix} \tag{5.19}$$

that depends on the initial error $\boldsymbol{z}^{(0)}$. Using the identity $\sum_{k=0}^{N-1} \boldsymbol{H}^k = (\boldsymbol{I} - \boldsymbol{H}^N)(\boldsymbol{I} - \boldsymbol{H})^{-1}$ and noting that[2]

$$(\boldsymbol{I} - \boldsymbol{H})^{-1} = \begin{pmatrix} (\boldsymbol{I} - \boldsymbol{Q})^{-1} & (\boldsymbol{I} - \boldsymbol{Q})^{-1}\boldsymbol{D}(\boldsymbol{I} - \boldsymbol{P})^{-1} \\ \boldsymbol{0} & (\boldsymbol{I} - \boldsymbol{P})^{-1} \end{pmatrix},$$

[2]For now, we presume that $\boldsymbol{I} - \boldsymbol{H}$ is invertible, a fact we will prove in the next section.

we can simplify (5.19) and get

$$\begin{pmatrix} \mathbb{E}[\boldsymbol{z}^{(N)}]^{\otimes 2} \\ \mathbb{E}\,\boldsymbol{z}^{(N)} \end{pmatrix} = \boldsymbol{H}^N \begin{pmatrix} [\boldsymbol{z}^{(0)}]^{\otimes 2} - \boldsymbol{v}_1 \\ \boldsymbol{z}^{(0)} - \boldsymbol{v}_2 \end{pmatrix} + \begin{pmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{pmatrix}. \tag{5.20}$$

The MSE can be expressed in terms of the vectorized second-moment matrix as

$$\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 = \mathrm{vec}(\boldsymbol{I}_n)^T \left(\mathbb{E}[\boldsymbol{z}^{(N)}]^{\otimes 2}\right). \tag{5.21}$$

Combining this with (5.20) yields the desired result. □

5.1.4 Error floor

When the noise vector is nonzero, the MSE in (5.18) will converge to a limiting value (*i.e.*, an error floor) that only depends on the error vector $\boldsymbol{\eta}$. To show this, we must show that the matrix $\boldsymbol{H}$ is a contraction, *i.e.* $\lambda_{\max}(\boldsymbol{H}) < 1$, where $\lambda_{\max}(\cdot)$ is the largest eigenvalue. We start with the following proposition:

Proposition 5.3. *The matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ satisfy $\lambda_{\max}(\boldsymbol{Q}) \leq \lambda_{\max}(\boldsymbol{P})$.*

Proof. We make use of the fact that the eigenvalues of $\boldsymbol{A} \otimes \boldsymbol{B}$ are all of the form $\lambda_i \mu_j$, where λ_i is an eigenvalue of $\boldsymbol{A}$ and μ_j is an eigenvalue of $\boldsymbol{B}$, as well as the bilinearity of the Kronecker product operator. Consider the following decomposition

of $\boldsymbol{Q}$:

$$\begin{aligned}\boldsymbol{Q} &= \sum_i p_i(\boldsymbol{P}_i \otimes \boldsymbol{P}_i) \\ &= \sum_i p_i(\boldsymbol{I} \otimes \boldsymbol{P}_i) - \sum_i p_i[(\boldsymbol{I} - \boldsymbol{P}_i) \otimes \boldsymbol{P}_i] \\ &= \boldsymbol{I} \otimes \boldsymbol{P} + \sum_i p_i[(\boldsymbol{P}_i - \boldsymbol{I}) \otimes \boldsymbol{P}_i]\end{aligned} \tag{5.22}$$

We can see that $\lambda_{\max}(\boldsymbol{I} \otimes \boldsymbol{P}) = \lambda_{\max}(\boldsymbol{P})$. Indeed, all the eigenvalues are the same, but with n times the multiplicity). Meanwhile, for any i, $\lambda_{\max}[(\boldsymbol{P}_i - \boldsymbol{I}) \otimes \boldsymbol{P}_i] \leq 0$ since $\boldsymbol{P}_i$ is positive semidefinite and $\lambda_{\max}(\boldsymbol{P}_i) \leq 1$. Using the sublinearity of the $\lambda_{\max}(\cdot)$ function, we have

$$\lambda_{\max}(\boldsymbol{Q}) \leq \lambda_{\max}(\boldsymbol{P}) + \sum_i p_i \lambda_{\max}((\boldsymbol{P}_i - \boldsymbol{I}) \otimes \boldsymbol{P}_i) \tag{5.23}$$

$$\leq \lambda_{\max}(\boldsymbol{P}), \tag{5.24}$$

and the proposition is proved. □

So we have that $\lambda_{\max}(\boldsymbol{Q}) \leq \lambda_{\max}(\boldsymbol{P}) < 1$. The matrix $\boldsymbol{P}$ is a contraction because $\boldsymbol{A}$ is overdetermined. We can show that $\boldsymbol{H}$ is a contraction as well through the following proposition

Proposition 5.4. *If the eigenvalues of $\boldsymbol{P}$ (including multiplicity) are $\lambda_1(\boldsymbol{P}), \ldots, \lambda_n(\boldsymbol{P})$ and the eigenvalues of $\boldsymbol{Q}$ are $\lambda_1(\boldsymbol{Q}), \ldots, \lambda_{n^2}(\boldsymbol{Q})$, then the eigenvalues of $\boldsymbol{H}$ are simply $\lambda_1(\boldsymbol{P}), \ldots, \lambda_n(\boldsymbol{P}), \lambda_1(\boldsymbol{Q}), \ldots, \lambda_{n^2}(\boldsymbol{Q})$.*

Proof. $\boldsymbol{P}$ or $\boldsymbol{Q}$ are symmetric so they can be diagonalized as $\boldsymbol{P} = \boldsymbol{S_P}\boldsymbol{\Lambda_P}\boldsymbol{S_P}^{-1}$ and $\boldsymbol{Q} = \boldsymbol{S_Q}\boldsymbol{\Lambda_Q}\boldsymbol{S_Q}^{-1}$. Here, $\boldsymbol{S_P}$ and $\boldsymbol{S_Q}$ have as columns the eigenvectors of $\boldsymbol{P}$ and $\boldsymbol{Q}$, respectively, while $\boldsymbol{\Lambda_P}$ and $\boldsymbol{\Lambda_Q}$ are diagonal matrices containing the eigenvalues.

Now, consider the following decomposition of $\boldsymbol{H}$:

$$\boldsymbol{H} = \begin{pmatrix} \boldsymbol{S_P} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S_Q} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Lambda_P} & \boldsymbol{S_P}^{-1}\boldsymbol{D}\boldsymbol{S_Q} \\ \boldsymbol{0} & \boldsymbol{\Lambda_Q} \end{pmatrix} \begin{pmatrix} \boldsymbol{S_P}^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S_Q}^{-1} \end{pmatrix}. \tag{5.25}$$

This shows that $\boldsymbol{H}$ is similar to (and thus has the same eigenvalues as)

$$\begin{pmatrix} \boldsymbol{\Lambda_P} & \boldsymbol{S_P}^{-1}\boldsymbol{D}\boldsymbol{S_Q} \\ \boldsymbol{0} & \boldsymbol{\Lambda_Q} \end{pmatrix}. \tag{5.26}$$

Since this is an upper triangular matrix, its eigenvalues can be read off the diagonal. These are simply the eigenvalues of $\boldsymbol{P}$ and $\boldsymbol{Q}$. □

Combining this with the previous proposition, we have $\lambda_{\max}(\boldsymbol{H}) = \lambda_{\max}(\boldsymbol{P})$. So $\boldsymbol{H}$ is a contraction. Most notably, $(\boldsymbol{I} - \boldsymbol{H})$ is invertible, as we presumed before. In particular, we can compute the inverse as

$$(\boldsymbol{I} - \boldsymbol{H})^{-1} = \begin{pmatrix} (\boldsymbol{I} - \boldsymbol{Q})^{-1} & (\boldsymbol{I} - \boldsymbol{Q})^{-1}\boldsymbol{D}(\boldsymbol{I} - \boldsymbol{P})^{-1} \\ \boldsymbol{0} & (\boldsymbol{I} - \boldsymbol{P})^{-1} \end{pmatrix}.$$

As $k \to \infty$, the first term of (5.18) vanishes, and so we approach an error floor:

$$\lim_{k\to\infty} \mathbb{E}\left\|\boldsymbol{z}^{(k)}\right\|^2 = \operatorname{vec}(\boldsymbol{I})^T \boldsymbol{v}_1 \tag{5.27}$$

$$= \operatorname{vec}(\boldsymbol{I})^T (\boldsymbol{I} - \boldsymbol{Q})^{-1}(\boldsymbol{e} + \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{P})^{-1}\boldsymbol{f}). \tag{5.28}$$

5.1.5 Average over the noise

Our exact MSE expression (5.18) given in Proposition 5.2 depends on the noise vector $\boldsymbol{\eta}$. In practice, of course, $\boldsymbol{\eta}$ is unknown, but we may have information about its statistics. In this section, we suppose that $\boldsymbol{\eta}$ is drawn from a probability distribution: in particular, we assume that its elements η_i are i.i.d. random variables with zero-mean and variance σ^2. Here, it is important to differentiate between two sources of randomness: the random row-selections made by the algorithm and the random vector $\boldsymbol{\eta}$. In what follows, $\mathbb{E}$ is understood as the conditional expectation operator over the randomness of the algorithm, with $\boldsymbol{\eta}$ fixed, and we define $\mathbb{E}_{\boldsymbol{\eta}}$ as the average over the noise.

It is convenient to rewrite (5.18) as

$$\begin{aligned}\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 = \operatorname{vec}(\boldsymbol{I}_n)^T \Big[\boldsymbol{Q}^N \left([\boldsymbol{z}^{(0)}]^{\otimes 2} - \boldsymbol{v}_1\right) \\ + f_N(\boldsymbol{D})\left(\boldsymbol{z}^{(0)} - \boldsymbol{v}_2\right)\Big] + \operatorname{tr}\operatorname{mat}(\boldsymbol{v}_1)\ ,\end{aligned} \tag{5.29}$$

where

$$f_N(\boldsymbol{D}) = \sum_{0\le k<N} \boldsymbol{Q}^k \boldsymbol{D} \boldsymbol{P}^{N-1-k}. \tag{5.30}$$

Since $f_N(\boldsymbol{D})$ is a linear function, we have $\mathbb{E}_{\boldsymbol{\eta}} f_N(\boldsymbol{D}) = f_N(\mathbb{E}_{\boldsymbol{\eta}}\boldsymbol{D}) = 0$. Averaging (5.29) over the noise, we get the following proposition.

Proposition 5.5. *The MSE of RKA at the Nth iteration averaged over both the ran-*

domness of the algorithm and noise is

$$\mathbb{E}_{\boldsymbol{\eta}}\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 = \mathrm{vec}(\boldsymbol{I}_n)^T\left[\boldsymbol{Q}^N\left([\boldsymbol{z}^{(0)}]^{\otimes 2} - \mathbb{E}_{\boldsymbol{\eta}}\boldsymbol{v}_1\right) - \mathbb{E}_{\boldsymbol{\eta}} f_N(\boldsymbol{D})\boldsymbol{v}_2\right] + \mathrm{tr}\,\mathrm{mat}(\mathbb{E}_{\boldsymbol{\eta}}\boldsymbol{v}_1). \tag{5.31}$$

This formula involves two noise-related quantities, $\mathbb{E}_{\boldsymbol{\eta}}\boldsymbol{v}_1$ and $\mathbb{E}_{\boldsymbol{\eta}} f_N(\boldsymbol{D})\boldsymbol{v}_2$, both of which are second-order in the noise. This shows that our knowledge of the second-order statistics of the noise is sufficient to compute them. In particular, the first term is given by

$$\mathbb{E}_{\boldsymbol{\eta}}\boldsymbol{v}_1 = \sigma^2(\boldsymbol{I}-\boldsymbol{Q})^{-1}\left[\sum_i p_i \frac{\widetilde{\boldsymbol{a}}_i^{\otimes 2}}{||\boldsymbol{a}_i||^2} + g\left((\boldsymbol{I}-\boldsymbol{P})^{-1}\right)\right],$$

where we define the matrix function

$$g(\boldsymbol{M}) = \sum_i \frac{p_i^2}{||\boldsymbol{a}_i||^2}\left(\widetilde{\boldsymbol{a}}_i \otimes \boldsymbol{P}_i + \boldsymbol{P}_i \otimes \widetilde{\boldsymbol{a}}_i\right)\boldsymbol{M}\widetilde{\boldsymbol{a}}_i. \tag{5.32}$$

(In these expressions the extra factors of $||\boldsymbol{a}_i||^2$ are not erroneous—they account for the varying signal-to-noise ratio of the measurements.)

The second noise-related term is computed by

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\eta}} f_N(\boldsymbol{D})\boldsymbol{v}_2 &= \sum_{0\le n<N} \boldsymbol{Q}^k \mathbb{E}_{\boldsymbol{\eta}}\boldsymbol{D}\boldsymbol{P}^{N-1-k}(\boldsymbol{I}-\boldsymbol{P})^{-1}\boldsymbol{f} \\ &= \sigma^2 \sum_{0\le k<N} \boldsymbol{Q}^k \mathbb{E}_{\boldsymbol{\eta}} g(\boldsymbol{P}^{N-1-k}(\boldsymbol{I}-\boldsymbol{P})^{-1}).\end{aligned}$$

Remark: The first term on the right-hand side of (5.31) decays exponentially because $\lambda_{\max}(\boldsymbol{Q}) < 1$. Thus, the limiting MSE averaged over both the randomness of the algorithm and noise is $\mathbb{E}_{\boldsymbol{\eta}}\mathbb{E}\left\|\boldsymbol{z}^{(\infty)}\right\|^2 = \mathrm{tr}\,\mathrm{mat}(\mathbb{E}_{\boldsymbol{\eta}}\boldsymbol{v}_1)$.

5.2 Error Exponents: Annealed vs. Quenched

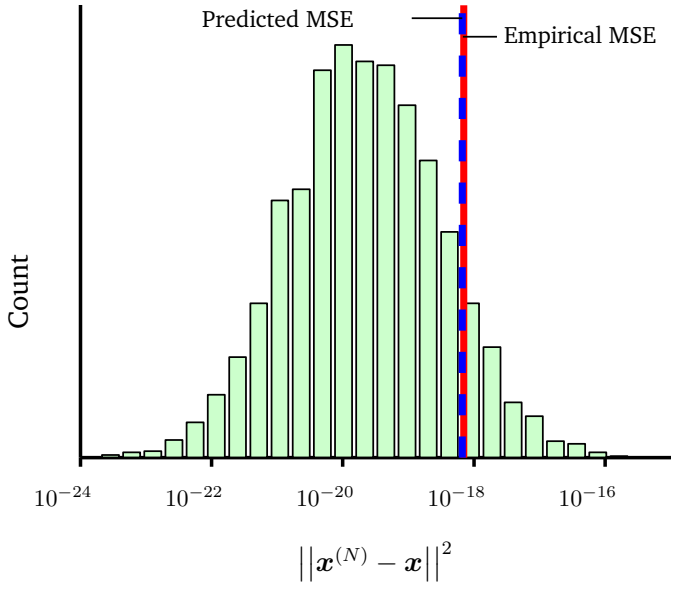
Proposition 5.1 confirms earlier bounds showing that the error decays exponentially in the noiseless case. In fact, for generic values of the initial error vector, we have $\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 = \exp(-\gamma_a N + o(N))$, where γ_a is the *annealed* error exponent, defined by

$$\gamma_a \stackrel{\text{def}}{=} \lim_{N\to\infty} -\frac{1}{N}\log \mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2. \tag{5.33}$$

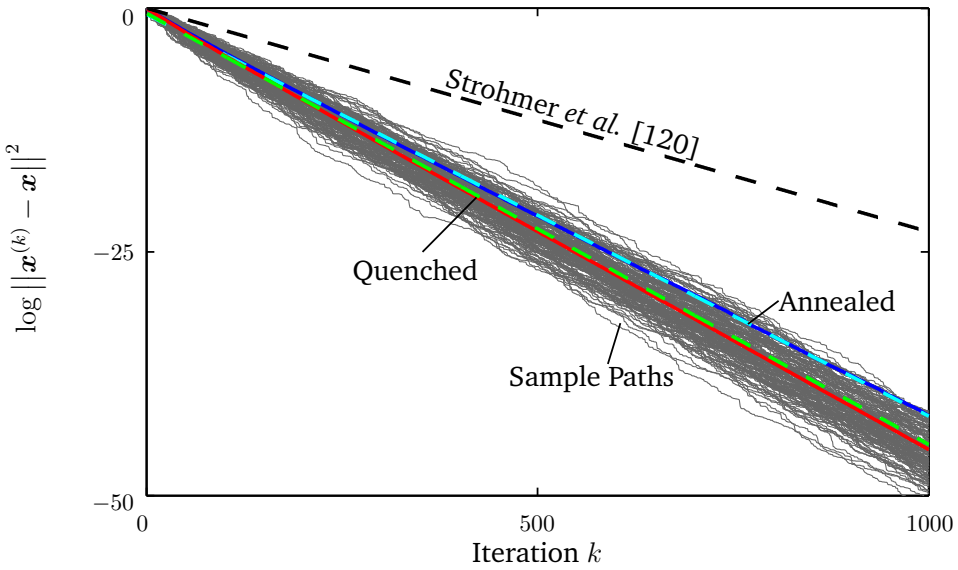
It is not hard to see that $\gamma_a = -\log \lambda_{\max}(\boldsymbol{Q})$, where $\lambda_{\max}(\cdot)$ is the largest eigenvalue of a matrix.

To test our result, we simulated 3007 trials of the Kaczmarz algorithm for solving a linear system of dimension 150×20. The same system was used for each run, as well as the same initial vector. The matrix $\boldsymbol{A}$ was chosen to have independent standard normal entries (note that none of our analysis depends on $\boldsymbol{A}$ being drawn in this way, and similar results can be obtained with other matrices). We tracked the error after every iteration for each run. The row was chosen uniformly at random for each iteration. Figure 5.1(a) shows a histogram of the errors after 1000 iterations. The histogram was computed and is plotted on a logarithmic scale because of the wide range of resulting errors. The empirical MSE is overlaid on the histogram, as well as our prediction based on Proposition 5.1. It is clear that our prediction matches the empirical value quite well. However, it is also clear that there is more to the story. Over 90% of the realizations have an error smaller than the mean,

Figure 5.1: (a) Histogram of squared errors after the simulation described in Section 5.2. The errors are plotted on a logarithmic scale to show the full range of errors; on a linear scale, the histogram is an L-shaped distribution with a spike at the origin and a long, thin tail. The location of the empirical MSE is overlaid on the histogram (red solid line), as is the exact MSE as given in Proposition 5.1 (blue dashed line). (b) Of the 3007 simulation trials, the "error trajectories" of 150 randomly-selected trials are plotted here (gray lines). On a logarithmic scale, there is a clear linear trend. Overlaid on these trajectories is the (annealed) average error trajectory (blue solid line) of all 3007 trials, and the prediction based on the annealed error exponent (cyan dashed line). We have also plotted the quenched average error trajectory, i.e. the average of the log of the error (red solid line), and the prediction based on the quenched error exponent (green dashed line) as given in (5.36). These are much more representative of the typical behavior of the algorithm. The upper bound of Strohmer *et al.* [120] is also shown (black dashed line).



(a)



(b)

Figure 5.1 (Continued).

which is more than 10^2 times smaller than the worst realization. It appears that the average error is not necessarily a great representation of the *typical* error; in reality, there are occasional, rare, extreme failures that cause the average error to be much higher than the "typical" error.

A more representative measure of the error's decay rate is the *quenched* error exponent:

$$\gamma_q \stackrel{\text{def}}{=} \lim_{N\to\infty} -\frac{1}{N}\mathbb{E}\log\|\boldsymbol{z}(N)\|^2 . \tag{5.34}$$

Here, the logarithm of the error is taken *before* the expectation. The annealed and quenched error exponents we have defined are formally similar to Lyapunov exponents of products of random matrices, a problem well-studied by statistical physicists for use in modeling dynamical systems [37]. The terms "annealed" and "quenched" are borrowed from their analysis and have certain physical meanings, but to us they are just convenient names for two interesting quantities.

The quenched error exponent is far more difficult to analyze than the annealed one, a fact well known to the physicists [37, 126]. Jensen's inequality tells us that $\gamma_q \geq \gamma_a$. To obtain more information, physicists often rely on non-rigorous heuristics that are verified numerically or experimentally. One such heuristic is the replica method, which provides an approximation for the quenched Lyapunov exponent [37]. The physicists have their own intuition for this approximation, but our engineer's intuition is quite simple. The quintessential heavy-tailed distribution is the log-normal distribution. So let us assume that the error distribution is

$\|\boldsymbol{z}(N)\|^2 \sim \log\text{-}\mathcal{N}(N\mu, N\sigma^2)$. Then $\log\|\boldsymbol{z}(N)\|^2 \sim \mathcal{N}(N\mu, N\sigma^2)$. The log-normal assumption is supported by the histogram in Figure 5.1(a): the logarithm of the squared errors appear to follow a Gaussian distribution. The quenched error exponent is seen to be simply $\gamma_q = -\mu$. Now we need to compute the parameters of the distribution. Under these assumptions, $\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 = \exp(N[\mu + \frac{1}{2}\sigma^2])$ and $\mathbb{E}\left|\left|\boldsymbol{z}^{(N)}\right|\right|^4 = \exp(N[2\mu + 2\sigma^2])$. Solving this system of equations, we obtain:

$$\mu = \frac{1}{N}\left[2\log\mathbb{E}\left\|\boldsymbol{z}^{(N)}\right\|^2 - \frac{1}{2}\log\mathbb{E}\left|\left|\boldsymbol{z}^{(N)}\right|\right|^4\right]. \tag{5.35}$$

Thus, our approximation for the quenched error exponent is

$$\gamma_q \approx 2\gamma_a - \frac{1}{2}\gamma_a^{(2)}, \tag{5.36}$$

where

$$\gamma_a^{(2)} = \lim_{N\to\infty} -\frac{1}{N}\log\mathbb{E}\left|\left|\boldsymbol{z}^{(N)}\right|\right|^4. \tag{5.37}$$

To compute $\gamma_a^{(2)}$, we define

$$\boldsymbol{Q}^{(2)} = \sum_{i=1}^{m} p_i \left(\boldsymbol{P}_{\boldsymbol{a}_i} \otimes \boldsymbol{P}_{\boldsymbol{a}_i} \otimes \boldsymbol{P}_{\boldsymbol{a}_i} \otimes \boldsymbol{P}_{\boldsymbol{a}_i}\right), \tag{5.38}$$

and have

$$\gamma_a^{(2)} = -\log\lambda_{\max}\left(\boldsymbol{Q}^{(2)}\right). \tag{5.39}$$

$\boldsymbol{Q}^{(2)}$ is an $n^4 \times n^4$ matrix, but it can be applied in time $O(mn^4)$ instead of the naive $O(n^8)$. So finding the largest eigenvalue is not as complex as one might naively expect.

Figure 5.1(b) illustrates our argument and shows just how good the replica method approximation is. We have plotted, on a logarithmic scale, the error trajectory of many trials as the iterations proceeded. (Only 150 randomly-selected trials are shown to prevent the figure from getting too cluttered). We have also plotted the logarithm of the average error, which matches the linear trendline predicted by the annealed error exponent γ_a, and the average of the logarithm of the error trajectories, which matches the linear trendline predicted by our approximation for the quenched error exponent γ_q. The quenched values are clearly more representative of the typical performance of the algorithm than the annealed ones. The close match indicates that our approximation is valid. For comparison purposes, we have also plotted the upper bound provided by Strohmer *et al.* [120].

5.3 Optimal Row-Selection Probabilities

Given a matrix $\boldsymbol{A}$, we may wish to choose the row selection probabilities p_1, p_2,..., p_m that provide the fastest convergence. A tractable way to do this is to optimize the annealed error exponent γ_a, which measures the decay rate of the MSE. This is equivalent to the following optimization problem:

$$(p_1, \ldots, p_m) = \underset{\boldsymbol{p} \in \Delta^{n-1}}{\arg\min}\, \lambda_{\max}(\boldsymbol{Q}), \tag{5.40}$$

where Δ^{n-1} is the unit simplex in $\mathbb{R}^n$. The function $\lambda_{\max}(\boldsymbol{Q})$ is convex [22], as is the set Δ^{n-1}, so (5.40) is a convex optimization problem (more specifically, it is a

semidefinite programming problem). Thus, finding the optimal probability distribution $\boldsymbol{p}$ is quite tractable. Note that Dai *et al.* recently considered an optimized randomized Kaczmarz algorithm [38], in which the row-selection probabilities were chosen to optimize a bound on the MSE's decay rate. However, we optimize the *exact* decay rate of the MSE.

To illustrate the kind of improvement possible by optimizing the row selection probabilities, and develop some intuition on the optimum choice, we computed the optimal values for a matrix of size 300×3. The elements of the matrix were chosen as independent Gaussian random variables with a variance of 0.5; the columns had means 0.5, 1, and 2, respectively. We used the `cvx` convex optimization software package to compute the optimal row selection probabilities for this matrix [60, 61].

Since the problem is invariant to the scale and sign of each rows, each row in the matrix $\boldsymbol{A}$ can be represented as a point on the unit hemisphere. Thus, the matrix and row probabilities can be illustrated as in Figure 5.2 by plotting each row as a point on a 2D projection of a unit hemisphere. We used the Lambert equal-area projection, which is measure-preserving and therefore allows us to accurately visualize the sampling density everywhere in the space. The darker points represent rows that are selected with high probability in the optimal selection scheme; the lighter ones are selected with lower probability. We would expect an optimal scheme to choose rows that are far from any other rows with higher probability than rows that are in close proximity to many other rows, in order to reduce redundancy and cover

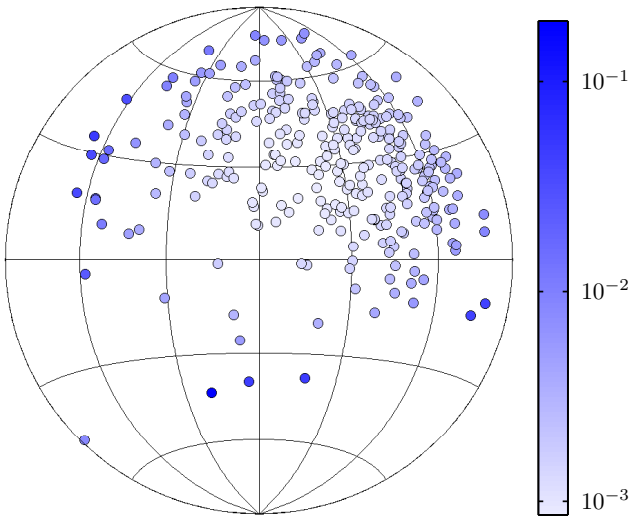


Figure 5.2: Optimal selection probabilities for a non-uniform matrix. The plot is an equal-area projection of the entire unit hemisphere in R^3. Each row in the matrix is represented by a point on the plot; the color represents the optimal selection probability computed using `cvx`.

the whole space. The figure conforms to this intuition.

Figure 5.3 illustrates the improvement of the optimal randomization scheme over simply choosing rows uniformly at random. After 20 iterations, the optimal scheme has an error 36 dB lower than the uniform scheme, and 12 dB lower than the sub-optimal scheme of Dai *et al.*

Of course, in practice, there is a tradeoff between the computation time saved by needing fewer iterations and the computation time spent determining the optimal row selection probabilities in advance. The main purpose of the exact optimization proposed in this work is to develop intuition and validate sub-optimal heuristics. A fast or on-line method for approximating the optimal probabilities would be very

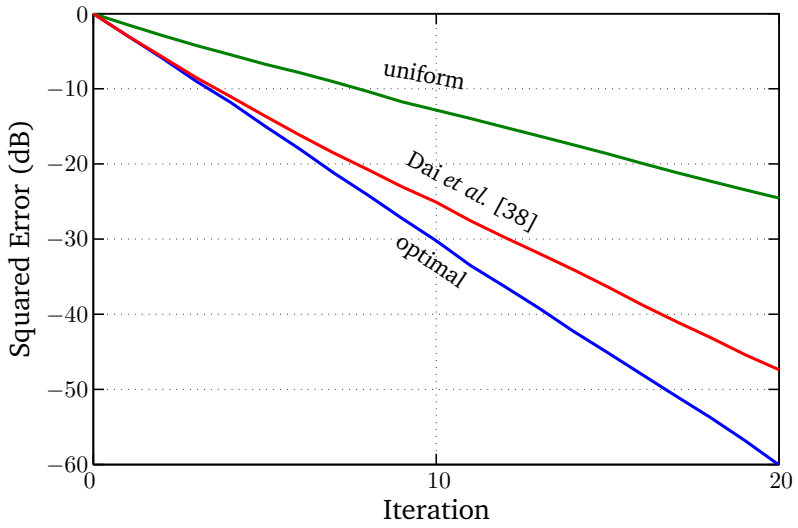


Figure 5.3: Quenched average squared errors versus RKA iteration under the uniform, Dai *et al.*'s approximate optimal, and optimal row selection probabilities, for the 1000 x 3 matrix described in the text.The average is taken over 1007 trials.

beneficial for large-scale problems.

5.4 Numerical Verification of the Noisy MSE formula

We also did simulations to verify the noisy MSE formulas (5.18) and (5.31). The results of two experiments are shown in this chapter. First, we tested the fixed noise formula (5.18). We drew a single noise vector $\boldsymbol{\eta}$ with $||\boldsymbol{\eta}||^2 = 1.6$, and a starting error $\boldsymbol{z}^{(0)}$, and choose a 150×50 measurement matrix $\boldsymbol{A}$ that had i.i.d. Gaussian entries. Then we ran 1007 separate trials of the randomized Kaczmarz algorithm, with each trial running for 2000 iterations and starting with an error vector $\boldsymbol{z}^{(0)}$. We plotted the average MSE of the trials at each iteration on a log scale. The results are

shown in Figure 5.4(a), and show that the expression we derived in (5.18) matches the numerical results very well. We also plotted existing bounds [94, 135] as well. The bounds are significantly higher than the true MSE.

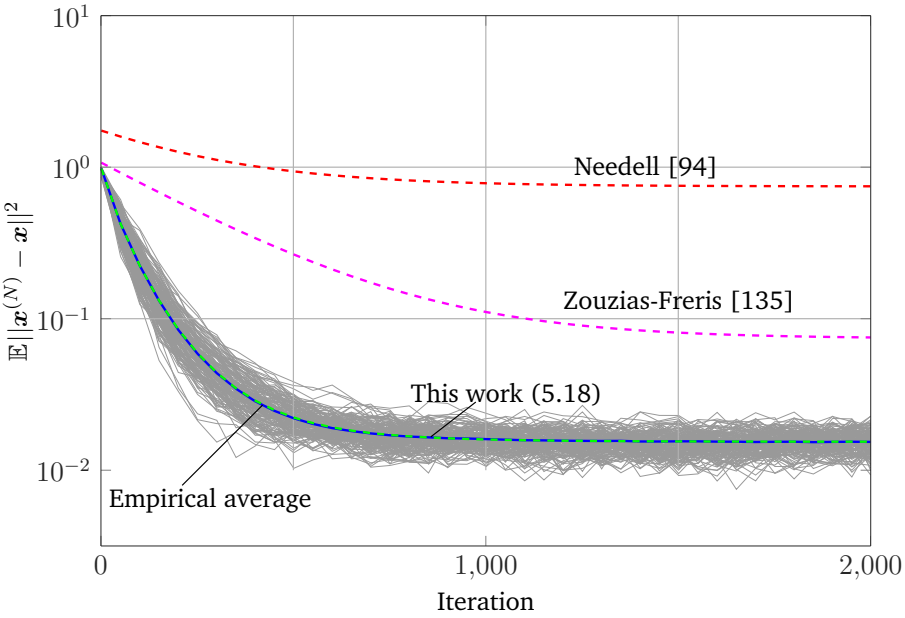
Next, we tested the noise-averaged formula (5.31). We used the AIR Tools package in MATLAB [63] to generate a tomography measurement matrix $\boldsymbol{A}$ of size 148×100. The noise vector $\boldsymbol{\eta}$ had i.i.d. entries with variance $\sigma^2 = 2.25 \times 10^{-4}$ and was drawn independently for each trial. We ran 1007 separate trials of the randomized Kaczmarz algorithm, with each trial running for 3000 iterations. The results are shown in Figure 5.4(b). The close match between empirical and theoretical curves verify our expression for the noise-averaged MSE (5.31). The graph also shows that the noise-averaged version of the Zouzias-Freris bound is more than two orders of magnitude higher than the true limiting MSE in this case.

5.5 Summary

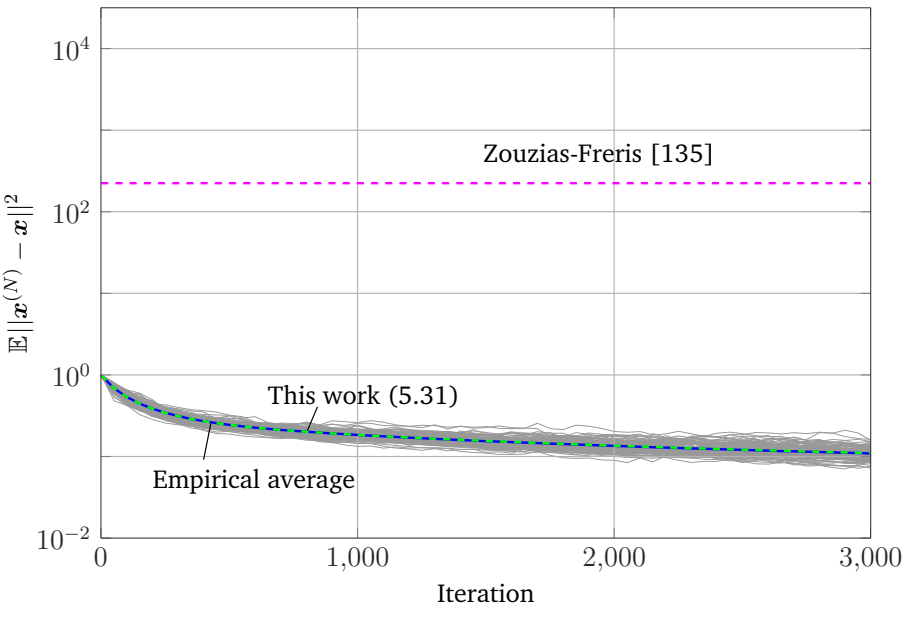
We provided a complete characterization of the randomized Kaczmarz algorithm. This included an exact formula for the MSE in both the noiseless and noisy cases, an expression for the limiting "error floor" in the noisy case, an exact expression for the annealed error exponent characterizing its decay rate in the noiseless case, plus an approximation for the quenched error exponent that captures the typical error decay rate. We also explored choosing the row-selection probabilities to

achieve the best convergence properties for the algorithm.

Figure 5.4: (a) The mean squared error $\mathbb{E}||\boldsymbol{x}^{(k)} - \boldsymbol{x}||^2$ is shown on a logarithmic scale as a function of the iteration number k. The matrix $\boldsymbol{A}$ has Gaussian entries, and the error vector η is fixed in advance, with $||\eta||^2 = 1.6$. The average results from 1007 trials are shown as the blue curve, and the results from 150 of the trials are shown in gray. The analytical expression (5.18) is shown as a dashed green line, and clearly matches the simulation results quite well. The Needell [94] and Zouzias-Freris [135] bounds are shown as well, and are far higher than the true MSE. (b) The mean square error $\mathbb{E}_{\boldsymbol{\eta}}\mathbb{E}||\boldsymbol{x}^{(k)} - \boldsymbol{x}||^2$ averaged over both the algorithm's randomness and the noise is shown on a logarithmic scale as a function of the iteration number k. The matrix $\boldsymbol{A}$ is the measurement matrix of a tomographic system (generated by the AIR Tools package [63]), and the error vector $\boldsymbol{\eta}$ is a zero mean Gaussian vector with variance 2.25×10^{-4}, drawn independently with each trial. The average of 1007 trials are shown in blue along with the results from 150 of the trials in gray. The analytical expression for the Gaussian noise case (5.31) clearly matches the simulation results. The noise-averaged Zouzias-Freris bound is shown as well for comparison.



(a)



(b)

Figure 5.4 (Continued).

Chapter 6

Conclusion

In this dissertation, we studied several problems in signal processing and inference on graphs. The main contributions of this work are:

- We introduced graph and spectral spread measurements for signals on graphs analogous to the time and frequency spreads of classical signal processing, and showed how to compute an "uncertainty curve" defining the tradeoff between the two. This tradeoff is directly analogous to the Heisenberg uncertainty principle in classical signal processing. The uncertainty curve can be computed parametrically by solving a sequence of eigenvalue problems. We also uncovered an intriguing connection to the graph heat equation: solutions to this equation seem to lie very close to the uncertainty curve.

- We analyzed the problem of detecting a random walk on a graph. We showed how to use statistical physics techniques to lower bound the error exponent for

this problem, revealing a phase transition. This provides a nice rule of thumb for deciding if the problem is tractable: when the SNR is greater than twice the entropy rate of the random walk, strong performance and exponential error decay is guaranteed.

- We introduced a technique for inferring the origin of an epidemic on a known network given a sequence of observations from a fraction of the nodes. We developed a fast Monte Carlo technique to simulate the epidemic from each possible source, extracted statistics from those simulations, and used them to approximate the likelihood. By maximizing this approximate likelihood, we found an estimate for the origin of the epidemic.

- We introduced a logistic autoregressive model (ALARM), a simple but very versatile model for binary dynamic processes on graphs. This model can capture a wide variety of behavior, including diffusion, negative influence of neighbors, and "stickiness" of a node's value. We showed how to use ℓ_1-regularized logistic regression to infer the graph structure from observations from this model. We also showed that the model is general enough that graph structure can be inferred even when the true data generation model is an epidemic model and not ALARM itself.

- We developed the first exact MSE formula for the randomized Kaczmarz algorithm, a linear system solver especially suited for sparse matrices often en-

countered in graph problems. The formula works in the noiseless and noisy cases. In the noisy case, we computed the limiting "error floor" that the algorithm converges to. In the noiseless case, we studied the annealed and quenched error exponents, which provide measures of the rate of decay of the MSE. We showed that the row selection probabilities could be chosen to optimize the annealed error exponent and improve performance. We showed numerically that existing bounds on the MSE are very loose, meaning that our expressions are a big improvement over the state-of-the-art performance bounds.

6.1 Future work

6.1.1 Spectral graph uncertainty principle

Although our basic analysis of the uncertainty curve works with an arbitrary distance function $d(\cdot, \cdot)$ on the graph, our results primarily focused on the geodesic distance, which is the smallest number of links in a path from one node to another. Extending these results to weighted graphs, in which each edge is associated with a weight capturing the strength of the connection, turns out to be nontrivial. Recently, a group of researchers analyzed how to do this [100]. One of the authors of that work also studied the influence of the clustering coefficient, a standard metric for the topology of a graph, on the shape of the uncertainty curves [105].

Another interesting line of investigation would be to fully understand the connection between the graph heat equation's solutions and the uncertainty curve. Is there a distance function under which the solutions follow the curve exactly? Can the difference between the uncertainty curve and the curve traced out by the solutions be bounded? An answer to these questions could lead to nicely localized signal representations computed easily by evolving the heat equation—perhaps something like the existing diffusion wavelet transform [34].

6.1.2 Detection of random walks

An important open problem is the development of an upper bound for the error exponent, particularly one with a similar phase transition as the lower bound. This would allow us to definitively say that below a certain threshold SNR, the detection problem will be impossible. We believe that such a nontrivial upper bound would require taking the asymptotic limit $M \to \infty$ of growing system size either simultaneously or after $N \to \infty$. This would require some sort of model for the growing graphs. For example, we may be able to model a K-regular graph with M nodes with K/M fixed as $M \to \infty$.

6.1.3 ALARM model

We conjecture that in the limit as the number of observations grows without bound, the ℓ^1-regularized logistic regression estimator can consistently estimate the

graph structure underlying the ALARM model. This parallels similar consistency results for the Gaussian case. A proof would provide strong backing for this technique in learning graphical structure collections of binary time series.

Using the concept of Granger causality, learning the parameters of the ALARM model can also help decide the existence and direction of causality in a network.

6.1.4 Randomized Kaczmarz

Our exact MSE formula for the randomized Kaczmarz algorithm is quite complicated. Existing bounds are very simple to compute, but we showed that they are often very loose. Perhaps our exact formula could be used to inspire simple, but tighter bounds on the performance.

6.2 Outlook

The field of signal processing and inference on graphs is fertile ground for interdisciplinary approaches to solving challenging new problems being raised by modern data sets. In this dissertation, we used concepts from time-frequency analysis, classical signal processing, decision theory, statistical physics, epidemiology, regression analysis, and more, to formulate, analyze, and solve problems involving data with network structure. This sort of broad-ranging synthesis will make the field very challenging—and fascinating—for years to come.

Appendix A

Spectral Graph Uncertainty Principle Proofs

A.1 The convexity of D

We would like to prove that the set $\boldsymbol{D}$ is convex as long as the number of vertices $N \geq 3$. (The need for such a condition will be made clear shortly.) This is equivalent to showing the following result.

Proposition A.1. *Suppose that there exist two vectors $\boldsymbol{x}_1, \boldsymbol{x}_2$ in $\mathbb{R}^N$ with $N \geq 3$, such that*

$$\boldsymbol{x}_i^T \boldsymbol{x}_i = 1,\ \boldsymbol{x}_i^T \boldsymbol{L} \boldsymbol{x}_i = s_i,\ \text{and}\ \boldsymbol{x}_i^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x}_i = g_i,\ \text{for}\ i = 1, 2. \tag{A.1}$$

Then for any $\beta \in [0, 1]$, we can always find a vector $\boldsymbol{x}$ in $\mathbb{R}^N$ satisfying

$$\boldsymbol{x}^T \boldsymbol{x} = 1, \quad \boldsymbol{x}^T \boldsymbol{L} \boldsymbol{x} = s,\ \text{and}\ \boldsymbol{x}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x} = g, \tag{A.2}$$

where $s \overset{def}{=} \beta s_1 + (1-\beta)s_2$ *and* $g \overset{def}{=} \beta g_1 + (1-\beta)g_2$.

We will prove the above proposition by recasting the problem in Sym_N, the Hilbert space of real, symmetric $N \times N$ matrices. The space is endowed with the Hilbert-Schmidt inner product defined by $\langle \boldsymbol{A}, \boldsymbol{B} \rangle_{\text{HS}} \overset{\text{def}}{=} \mathrm{tr}(\boldsymbol{A}^T \boldsymbol{B}) = \mathrm{tr}(\boldsymbol{A}\boldsymbol{B})$, where tr$(\cdot)$ is the trace of a matrix. Every $\boldsymbol{x} \in \mathbb{R}^N$ can be mapped onto a matrix $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T$ in Sym_N. Finding a vector $\boldsymbol{x}$ satisfying the conditions in (A.2) then boils down to finding a rank-one positive semidefinite matrix $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T$ satisfying the following three constraints

$$\mathrm{tr}(\boldsymbol{X}) = 1, \quad \mathrm{tr}(\boldsymbol{L}\boldsymbol{X}) = s \quad \text{and } \mathrm{tr}(\boldsymbol{P}_{u_0}^2 \boldsymbol{X}) = g. \tag{A.3}$$

The requirement that $\boldsymbol{X}$ be a rank-one matrix makes this a hard problem, because the cone of rank-one matrices is not convex. Instead, we will use the following theorem to relax the problem to the cone of positive semidefinite matrices $\mathcal{S}_+^N$, which is convex.

Theorem A.1 (Barvinok [15])**.** *Suppose that* $R > 0$ *and* $N \geq R + 2$*. Let* $\mathcal{H} \subset \mathrm{Sym}_N$ *be an affine subspace such that* $\mathrm{codim}(\mathcal{H}) \leq \binom{R+2}{2}$*. If the intersection* $\mathcal{S}_+^N \cap \mathcal{H}$ *is nonempty and bounded, then there is a matrix* $\boldsymbol{X}$ *in* $\mathcal{S}_+^N \cap \mathcal{H}$ *such that* $\mathrm{rank}(\boldsymbol{X}) \leq R$*.*

Proof of Proposition A.1. First, we note that the three equalities in (A.3) are all affine constraints on $\boldsymbol{X}$. Together, they define a hyperplane $\mathcal{H} \subset \mathrm{Sym}_N$ with $\mathrm{codim}(\mathcal{H}) \leq 3 = \binom{1+2}{2}$. (In fact, $\boldsymbol{I}$, $\boldsymbol{L}$, and $\boldsymbol{P}_{u_0}^2$ are linearly independent, so

$\operatorname{codim}(\mathcal{H}) = 3$.) To apply Theorem A.1, we verify next that $\mathcal{S}_+^N \cap \mathcal{H}$ is nonempty and bounded.

First we show that it is bounded: let $\boldsymbol{X}$ be an arbitrary matrix in the intersection $\mathcal{S}_+^N \cap \mathcal{H}$ (assuming one exists), and let $\{\nu_1, \nu_2, \ldots, \nu_N\}$ be its eigenvalues. The equalities $1 = \operatorname{tr}(\boldsymbol{X}) = \sum_{n=1}^N \nu_n$, together with the nonnegativity of the eigenvalues, imply that

$$\|\boldsymbol{X}\|_{\text{HS}}^2 = \operatorname{tr}(\boldsymbol{X}^2) = \sum_{n=1}^N \nu_n^2 \leq \sum_{n=1}^N \nu_n = 1.$$

Therefore, $\mathcal{S}_+^N \cap \mathcal{H}$ is a subset of the unit ball in Sym_N and is thus bounded.

To show that $\mathcal{S}_+^N \cap \mathcal{H}$ is nonempty, we explicitly construct a member of the set. Let $\boldsymbol{x}_1, \boldsymbol{x}_2$ be the two vectors satisfying (A.1). On mapping the vectors to two matrices $\boldsymbol{X}_1 \overset{\text{def}}{=} \boldsymbol{x}_1 \boldsymbol{x}_1^T$ and $\boldsymbol{X}_2 \overset{\text{def}}{=} \boldsymbol{x}_2 \boldsymbol{x}_2^T$, the constraints in (A.1) can be rewritten as

$$\operatorname{tr}(\boldsymbol{X}_i) = 1,\ \operatorname{tr}(\boldsymbol{L}\boldsymbol{X}_i) = s_i \text{ and } \operatorname{tr}(\boldsymbol{P}_{u_0}^2 \boldsymbol{X}_i) = g_i, \text{ for } i = 1, 2.$$

$\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are both in $\mathcal{S}_+^N$. Now set $\boldsymbol{X}' = \beta \boldsymbol{X}_1 + (1-\beta)\boldsymbol{X}_2$. It is easy to see that $\boldsymbol{X}' \in \mathcal{H}$ and, because $\mathcal{S}_+^N$ is convex, $\boldsymbol{X}' \in \mathcal{S}_+^N$ as well. To be sure, the matrix $\boldsymbol{X}' \in \mathcal{S}_+^N \cap \mathcal{H}$ is not necessarily of rank one. However, the result of Theorem A.1 (for the case when $R = 1$) guarantees the existence of a rank one matrix $\boldsymbol{X}$ in $\mathcal{S}_+^N \cap \mathcal{H}$. Decomposing this matrix as $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T$ and using the equivalence between (A.2) and (A.3), we can conclude that the resulting vector $\boldsymbol{x}$ satisfies all the constraints in (A.2). □

Remark: The above proof uses Theorem A.1 for the case when $R = 1$. Conse-

quently, we need to work with $N \geq R + 2 = 3$. This requirement is sharp in that the achievable region $\boldsymbol{D}$ for a graph with two vertices (*i.e.*, $N = 2$) is not convex. The only connected graph with $N = 2$ is the complete graph. All unit-norm signals on this graph can be parametrized as $(\cos\theta, \sin\theta)$. By computing the corresponding graph Laplacian and distance matrices, it is easy to show that the achievable region is only the boundary of an ellipse (not including its interior) and hence is not convex.

A.2 Proof of Lemma 2.1

(a) For any $\alpha_1 < \alpha_2$, let $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ be two unit-norm eigenvectors in $\mathcal{S}(\alpha_1)$ and $\mathcal{S}(\alpha_2)$, respectively. Applying Rayleigh's inequality, we get $\boldsymbol{v}_2^T \boldsymbol{M}(\alpha_1)\boldsymbol{v}_2 \geq q(\alpha_1) = \boldsymbol{v}_1^T \boldsymbol{M}(\alpha_1)\boldsymbol{v}_1$ Similarly, we have $-\boldsymbol{v}_2^T \boldsymbol{M}(\alpha_2)\boldsymbol{v}_2 \geq -\boldsymbol{v}_1^T \boldsymbol{M}(\alpha_2)\boldsymbol{v}_1$. A combination of these two inequalities leads to

$$\boldsymbol{v}_2^T \left(\boldsymbol{M}(\alpha_1) - \boldsymbol{M}(\alpha_2)\right) \boldsymbol{v}_2 \geq \boldsymbol{v}_1^T \left(\boldsymbol{M}(\alpha_1) - \boldsymbol{M}(\alpha_2)\right) \boldsymbol{v}_1. \tag{A.4}$$

Recall that $\boldsymbol{M}(\alpha) = \boldsymbol{P}_{u_0}^2 - \alpha \boldsymbol{L}$, and therefore $\boldsymbol{M}(\alpha_1) - \boldsymbol{M}(\alpha_2) = (\alpha_2 - \alpha_1)\boldsymbol{L}$. Replacing this identity into (A.4), we thus have

$$\boldsymbol{v}_2^T \boldsymbol{L} \boldsymbol{v}_2 \geq \boldsymbol{v}_1^T \boldsymbol{L} \boldsymbol{v}_1.$$

Note that $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ can be arbitrary unit-norm elements in $\mathcal{S}(\alpha_1)$ and $\mathcal{S}(\alpha_2)$, respectively. If, in particular, we choose $\boldsymbol{v}_1, \boldsymbol{v}_2$ to be those that attain the maximization in

(2.21), we get $h_+(\alpha_2) = \boldsymbol{v}_2^T \boldsymbol{L} \boldsymbol{v}_2 \geq \boldsymbol{v}_1^T \boldsymbol{L} \boldsymbol{v}_1 = h_+(\alpha_1)$. Similarly, we can show that $h_-(\alpha_2) \geq h_-(\alpha_1)$.

(b) We will only consider the limits when α tends to $-\infty$ as given in (2.23). The other case, when α tends to $+\infty$, can be analyzed in a similar way, and its proof will be omitted. Let $\alpha > 0$ be any positive number. By definition,

$$h_+(\alpha) \geq h_-(\alpha) \geq 0, \tag{A.5}$$

where the second inequality is due to the Laplacian matrix $\boldsymbol{L}$ being positive semidefinite. Next, we show that $h_+(\alpha)$ can be made arbitrarily close to 0 as $\alpha \to -\infty$. To that end, let $\boldsymbol{v}$ be any unit-norm eigenvector in $\mathcal{S}(\alpha)$, and $\boldsymbol{f}_1$ be the first eigenvector of $\boldsymbol{L}$ as defined in (2.4). Since $\mathcal{S}(\alpha)$ is associated with the smallest eigenvalue $q(\alpha)$, we have, from Rayleigh's inequality,

$$\boldsymbol{v}^T(\boldsymbol{P}_{u_0}^2 - \alpha \boldsymbol{L})\boldsymbol{v} \leq \boldsymbol{f}_1^T(\boldsymbol{P}_{u_0}^2 - \alpha \boldsymbol{L})\boldsymbol{f}_1 = \boldsymbol{f}_1^T \boldsymbol{P}_{u_0}^2 \boldsymbol{f}_1,$$

with the equality coming from the identity $\boldsymbol{L}\boldsymbol{f}_1 = 0$. For any $\alpha < 0$, rearranging the above expression leads to

$$\boldsymbol{v}^T \boldsymbol{L} \boldsymbol{v} \leq -\frac{1}{\alpha}\left(\boldsymbol{f}_1^T \boldsymbol{P}_{u_0}^2 \boldsymbol{f}_1 - \boldsymbol{v}^T \boldsymbol{P}_{u_0}^2 \boldsymbol{v}\right) \leq -\frac{\mathcal{E}_G^2(u_0)}{\alpha}, \tag{A.6}$$

where the second inequality uses the bound of the graph spread as provided in Proposition 2.1. Since (A.6) holds for *any* nonzero element $\boldsymbol{v}$ from $\mathcal{S}(\alpha)$, we must have $h_+(\alpha) \leq -\mathcal{E}_G^2(u_0)/\alpha$, which, when combined with (A.5), completes the proof.

(c) First, using eigenvalue perturbation results, we will derive a candidate set $\mathcal{A}$ of points such that $q(\alpha)$ is certainly analytic on $[a, b]\backslash\mathcal{A}$. We will show that $\mathcal{A}$ is finite,

so that the set of nonanalytic points of $q(\alpha)$ is finite as well. Then, we will compute $h_-(\alpha)$ and $h_+(\alpha)$ explicitly, and show that they are are left- and right-continuous, respectively, and that they are equal to the negative left- and right-derivatives of $q(\alpha)$, respectively. We will then show that $h_-(\alpha) = h_+(\alpha)$ everywhere except a subset $\mathcal{B} \subseteq \mathcal{A}$; therefore, they satisfy (2.24). Since $\mathcal{A}$ is finite, it follows that $\mathcal{B}$ is finite as well.

The starting point of our analysis is the following result.

Proposition A.2. *There exist N analytic functions $\lambda_1(\cdot), \ldots, \lambda_N(\cdot)$ and N analytic vector-valued functions $\boldsymbol{x}_1(\cdot), \ldots, \boldsymbol{x}_N(\cdot)$ such that*

$$\boldsymbol{M}(\alpha)\boldsymbol{x}_i(\alpha) = \lambda_i(\alpha)\boldsymbol{x}_i(\alpha), \tag{A.7}$$

and $\boldsymbol{x}_i(\alpha)^T\boldsymbol{x}_j(\alpha) = \delta_{ij}$.

Proof. Standard perturbation results [80, p. 404] guarantee the existence of such functions for any matrix function that is analytic and whose value is always Hermitian. The function $M(\cdot)$ as defined in (2.18) is affine in α, and thus analytic; it is symmetric and real for every α, and thus Hermitian. Therefore functions with the properties listed in the proposition do exist. □

From Proposition A.2, we can write $q(\alpha)$ as

$$q(\alpha) = \min_{1 \leq i \leq N} \lambda_i(\alpha), \tag{A.8}$$

where the $\{\lambda_i(\cdot)\}_i$ are the eigenvalue functions guaranteed by the proposition. For any $\alpha_0 \in \mathbb{R}$, if $S(\alpha_0)$ has dimension one, then precisely one of the eigenvalue functions is equal to $q(\cdot)$ at α_0, say $\lambda_k(\alpha_0) = q(\alpha_0)$. Pick some $\varepsilon < \frac{1}{2}\min_{j\neq k}|\lambda_j(\alpha_0) - \lambda_k(\alpha_0)|$. Since every $\lambda_j(\cdot)$ is analytic, we can find some neighborhood $\mathcal{N}$ of α_0 for which $|\lambda_j(\alpha) - \lambda_j(\alpha_0)| < \varepsilon$ for every j. This guarantees that $\lambda_k(\alpha) < \lambda_j(\alpha)$ on $\mathcal{N}$ for every $j \neq k$. Thus $q(\alpha) = \lambda_k(\alpha)$ on $\mathcal{N}$. Since $\lambda_k(\cdot)$ is analytic on $\mathcal{N}$, we have that $q(\cdot)$ is analytic on $\mathcal{N}$ and therefore at α_0. We can make this more general. Suppose instead of only one eigenvalue function attaining the minimum at α_0, there are multiple eigenvalue functions [*e.g.*, two, denoted by $\lambda_{k_1}(\cdot)$ and $\lambda_{k_2}(\cdot)$] that attain the minimum, and that they are all equal on a neighborhood $\mathcal{N}$ of α_0. All the other eigenvalue functions are larger at α_0. Again, the analyticity allows us to find a neighborhood $\mathcal{N}' \subseteq \mathcal{N}$ on which all the other eigenvalue functions are larger than $\lambda_{k_1}(\cdot) = \lambda_{k_2}(\cdot)$. Now, since $q(\alpha) = \lambda_{k_1}(\alpha) = \lambda_{k_2}(\alpha)$, the function $q(\alpha)$ is analytic on $\mathcal{N}'$ as well.

Thus, a necessary condition for $q(\cdot)$ to be nonanalytic at α_0 is that two (or more) *distinct* eigenvalue functions must intersect at α_0. Define $\mu_j(\cdot), j = 1, \ldots, N', N' \leq N$ as the set of distinct eigenvalue functions, and let n_j be the multiplicity of the eigenvalue function $\mu_j(\cdot)$. Now consider an arbitrary finite interval $[a, b]$ and define

$$\mathcal{A} = \bigcup_{1\leq i<j\leq N'} \{\alpha \in [a, b] : \mu_i(\alpha) = \mu_j(\alpha)\}.$$

It is a well known property of analytic functions that if they are equal on more than a finite set of points in an interval, then they are identical. Since the $\mu_i(\cdot)$ are

distinct analytic functions, no two of them can be equal on more than a finite set of points in $[a, b]$. Thus $\mathcal{A}$ is the finite union of finite sets, and therefore contains only a finite number of points

Next, we connect $q(\alpha)$ to $h_+(\alpha)$ and $h_-(\alpha)$. At any point $\alpha_0 \in [a, b]$, there can be $k \geq 1$ distinct eigenvalue functions that achieve the minimum in (A.8). Without loss of generality, we shall assume they are the first k functions, $\mu_1(\cdot), \ldots, \mu_k(\cdot)$. The associated eigenvectors, $\boldsymbol{x}_{ij}(\alpha_0)$, for $i = 1, \ldots, k$ and $j = 1, \ldots, n_i$, form an orthonormal basis for the eigenspace $\mathcal{S}(\alpha_0)$. Any unit-norm element $\boldsymbol{v} \in \mathcal{S}(\alpha_0)$ can then be written as $\boldsymbol{v} = \sum_{i=1}^{k} \sum_{j=1}^{n_i} c_{ij}\, \boldsymbol{x}_{ij}(\alpha_0)$, for some constant coefficients $\{c_{ij}\}$ satisfying $\sum_{i=1}^{k} \sum_{j=1}^{n_i} c_{ij}^2 = 1$.

We now define an analytic function $\boldsymbol{v}(\alpha) \stackrel{\text{def}}{=} \sum_{i=1}^{k} \sum_{j=1}^{n_i} c_{ij}\, \boldsymbol{x}_{ij}(\alpha)$, with $\boldsymbol{v}(\alpha_0) = \boldsymbol{v}$. The eigenvalue identity in (A.7) implies that

$$\boldsymbol{M}(\alpha)\boldsymbol{v}(\alpha) = \sum_{i=1}^{k} \sum_{j=1}^{n_i} c_{ij} \mu_i(\alpha) \boldsymbol{x}_{ij}(\alpha)$$

. Differentiating both sides of this equality yields

$$\boldsymbol{M}'(\alpha)\, \boldsymbol{v}(\alpha) + \boldsymbol{M}(\alpha)\boldsymbol{v}'(\alpha) = \tag{A.9}$$
$$\sum_{i=1}^{k} \sum_{j=1}^{n_i} c_{ij} \mu_i'(\alpha)\, \boldsymbol{x}_{ij}(\alpha) + \sum_{i=1}^{k} \sum_{j=1}^{n_i} c_{ij} \mu_i(\alpha) \boldsymbol{x}_{ij}'(\alpha).$$

Evaluating (A.9) at $\alpha = \alpha_0$, pre-multiplying it by $\boldsymbol{v}^T(\alpha_0)$ and using the substitutions

$\boldsymbol{M}'(\alpha) = -\boldsymbol{L}$, $\boldsymbol{M}(\alpha_0)\boldsymbol{v}(\alpha_0) = q(\alpha_0)\boldsymbol{v}(\alpha_0)$, and $\mu_i(\alpha_0) = q(\alpha_0)$ for every i, we get

$$-\boldsymbol{v}^T(\alpha_0)\boldsymbol{L}\boldsymbol{v}(\alpha_0) + q(\alpha_0)\boldsymbol{v}^T(\alpha_0)\boldsymbol{v}'(\alpha_0) = \tag{A.10}$$
$$\sum_{i=1}^{k}\sum_{j=1}^{n_i} c_{ij}^2\mu_i'(\alpha_0) + q(\alpha_0)\sum_{i=1}^{k}\sum_{j=1}^{n_i} c_{ij}\boldsymbol{v}^T(\alpha_0)\boldsymbol{x}_{ij}'(\alpha_0).$$

The second terms on the left-hand and right-hand sides of (A.10) are equal, leaving us with

$$\boldsymbol{v}^T(\alpha_0)\boldsymbol{L}\boldsymbol{v}(\alpha_0) = -\sum_{i=1}^{k}\sum_{j=1}^{n_i} c_{ij}^2\mu_i'(\alpha_0). \tag{A.11}$$

By definition, $h_+(\alpha_0)$ and $h_-(\alpha_0)$ are the two extreme values of $\boldsymbol{v}^T(\alpha_0)\boldsymbol{L}\boldsymbol{v}(\alpha_0)$. Maximizing (and minimizing) the quantity in (A.11) subject to the unit-norm constraint

$$\sum_{i=1}^{k}\sum_{j=1}^{n_i} c_{ij}^2 = 1,$$

we have

$$h_+(\alpha_0) = \max_{1\le i\le k}(-\mu_i'(\alpha_0)) \text{ and } h_-(\alpha_0) = \min_{1\le i\le k}(-\mu_i'(\alpha_0)). \tag{A.12}$$

Now, there must exist some $m, \ell \in \{1, \ldots, k\}$ such that

$$q(\alpha) = \begin{cases} \mu_\ell(\alpha) & \text{if } \alpha \le \alpha_0 \\ \mu_m(\alpha) & \text{if } \alpha \ge \alpha_0 \end{cases} \tag{A.13}$$

on some neighborhood $\mathcal{N}$ of α_0, which can be chosen to be small enough that $\mathcal{N} \cap \mathcal{A} = \{\alpha_0\}$ if $\alpha_0 \in \mathcal{A}$ or $\mathcal{N} \cap \mathcal{A} = \emptyset$ otherwise. We must have $\mu_m'(\alpha_0) = \min_{1\le i\le k}\mu_i'(\alpha_0)$, since if $\mu_j'(\alpha_0) < \mu_m'(\alpha_0)$ for some j, then on a sufficiently small neighborhood of α_0, we would have $q(\alpha) = \mu_j(\alpha) < \mu_m(\alpha)$ for $\alpha > \alpha_0$, contradicting

(A.13).[1] Meanwhile, away from α_0 there are no other points in $\mathcal{N}$ at which multiple distinct eigenvalue functions intersect. Thus, from (A.12), we have that $h_+(\alpha) = -\mu'_m(\alpha)$ on $\mathcal{N} \cap [\alpha_0, \infty)$. Since the $\mu_i(\cdot)$ are all analytic, $h_+(\alpha)$ is right-continuous at α_0. Furthermore, since $q(\alpha) = \mu_m(\alpha)$ on $\mathcal{N} \cap [\alpha_0, \infty)$, $h_+(\alpha_0)$ is equal to the negative right-derivative of $q(\alpha)$ at α_0. By similar arguments, we can show that $h_-(\alpha)$ is left-continuous at α_0 and is equal to the negative left-derivative of $q(\alpha)$ at α_0.

A necessary condition for $h_-(\alpha_0) \neq h_+(\alpha_0)$ is that $k > 1$, *i.e.*, there are multiple distinct eigenvalue functions achieving the minimum in (A.8). Thus the set of points $\mathcal{B}$ at which they differ satisfies $\mathcal{B} \subseteq \mathcal{A}$, so $\mathcal{B}$ is finite. Meanwhile, if $h_+(\alpha_0) = h_-(\alpha_0)$, then the equality must hold for all $\alpha \in \mathcal{N}$ as well because of the way we constructed the neighborhood $\mathcal{N}$. Since $h_-(\alpha)$ is left-continuous and $h_+(\alpha)$ is right-continuous at α_0, both functions are continuous at α_0. Equality also means the left- and right-derivatives of $q(\alpha)$ are equal at α_0, and thus $q'(\alpha_0)$ is well-defined with $h_+(\alpha_0) = h_-(\alpha_0) = -q'(\alpha_0)$.

[1] The requirement that $\mu'_m(\alpha_0) = \min_{1 \leq i \leq k} \mu'_i(\alpha_0)$ might not always be sufficient to uniquely determine m, however. In the case that multiple distinct eigenvalue functions achieve the minimum derivative, $\mu_m(\cdot)$ is then determined by comparing the higher order derivatives. This nuance does not affect our proof, which only depends on the first derivative.

A.3 Proof of Proposition 2.3

For $N = 2$ the proposition is trivial, so let us assume $N > 2$. By Theorem 2.1, $\widetilde{\boldsymbol{x}}$ must be an eigenvector associated with the smallest eigenvalue of $\boldsymbol{M}(\alpha) = \boldsymbol{P}_{u_0}^2 - \alpha \boldsymbol{L}$ for some α, where $\boldsymbol{L}$ and $\boldsymbol{P}_{u_0}$ are given by (2.27) and (2.28), respectively. $\boldsymbol{M}(\alpha)$ is given in block form as

$$\boldsymbol{M}(\alpha) = \left[\begin{array}{c|c} -\alpha & \frac{\alpha}{N-1}\mathbf{1}_{N-1}^T \\ \hline \frac{\alpha}{N-1}\mathbf{1}_{N-1} & \boldsymbol{B} \end{array}\right],$$

where $\boldsymbol{B}$ is the $(N-1) \times (N-1)$ circulant matrix

$$\boldsymbol{B} = \left(1 - \frac{N}{N-1}\alpha\right)\boldsymbol{I}_{N-1} + \frac{\alpha}{N-1}\mathbf{1}_{N-1}\mathbf{1}_{N-1}^T.$$

Let $\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{N-2}\}$ be an orthonormal set of vectors in $\mathbb{R}^{N-1}$ such that $\boldsymbol{w}_i \perp \mathbf{1}_{N-1}$. This set spans the subspace of vectors in $\mathbb{R}^{N-1}$ orthogonal to $\mathbf{1}_{N-1}$. It is easy to verify that $\boldsymbol{B}\boldsymbol{w}_i = (1 - \frac{N}{N-1}\alpha)\boldsymbol{w}_i$. Furthermore, if we set $\boldsymbol{v}_i = (0, \boldsymbol{w}_i^T)^T$, then we can see that $\boldsymbol{v}_i$ are all eigenvectors of $\boldsymbol{M}(\alpha)$ with eigenvalue $1 - \frac{N}{N-1}\alpha$.

If we can show that this is not the smallest eigenvalue of $\boldsymbol{M}(\alpha)$, *i.e.* that $q(\alpha) \neq 1 - \frac{N}{N-1}\alpha$, then it follows that $\widetilde{\boldsymbol{x}}$ [an eigenvector of $\boldsymbol{M}(\alpha)$ corresponding to $q(\alpha)$] must be orthogonal to every $\boldsymbol{v}_i$ for $i = 1, \ldots, N-2$. This will then guarantee that $\widetilde{\boldsymbol{x}}$ is of the form (2.29).

To show that $q(\alpha) \neq 1 - \frac{N}{N-1}\alpha$, we let $\boldsymbol{y} = [y_1, \ldots, y_N]$ be chosen such that $||\boldsymbol{y}|| = 1$, $y_1 \neq 0$, and $\boldsymbol{y}^T\mathbf{1}_N = 0$. This last property makes $\boldsymbol{y}$ an eigenvector of $\boldsymbol{L}$ with eigenvalue $\frac{N}{N-1}$. We have $\boldsymbol{y}^T\boldsymbol{P}_{u_0}^2\boldsymbol{y} = 1 - y_1^2$ and $\boldsymbol{y}^T\boldsymbol{L}\boldsymbol{y} = \frac{N}{N-1}$. Thus $\boldsymbol{y}^T\boldsymbol{M}(\alpha)\boldsymbol{y} =$

$1 - y_1^2 - \frac{N}{N-1}\alpha < 1 - \frac{N}{N-1}\alpha$. It follows from the Rayleigh inequality that $q(\alpha) \leq \boldsymbol{y}^T \boldsymbol{M}(\alpha)\boldsymbol{y} < 1 - \frac{N}{N-1}\alpha$, proving the proposition.

Remark: With small modifications, the above proof can be used to demonstrate that the same property holds for star graphs, *i.e.* any vector achieving the uncertainty curve must be of the form in (2.29). For a star graph with N vertices, we have

$$\boldsymbol{M}(\alpha) = \left[\begin{array}{c|c} -\alpha & \frac{\alpha}{\sqrt{N-1}}\mathbf{1}_{N-1}^T \\ \hline \frac{\alpha}{\sqrt{N-1}}\mathbf{1}_{N-1} & (1-\alpha)\boldsymbol{I}_{N-1} \end{array}\right]. \tag{A.14}$$

Again, there is an $(N-2)$–dimensional eigenspace spanned by the same set $\{\boldsymbol{v}_i\}_{i=1}^{N-2}$ as in the complete graph case above. In this case, the eigenvalue associated with that subspace is $1-\alpha$. Thus, to show that the smallest eigenvector is of the desired form, we must simply show that there is some unit norm vector $\boldsymbol{y}$ for which $\boldsymbol{y}^T\boldsymbol{M}(\alpha)\boldsymbol{y} < 1-\alpha$, guaranteeing that the eigenvector associated with the smallest eigenvalue is orthogonal to the eigenspace spanned by $\{\boldsymbol{v}_i\}_{i=1}^{N-1}$. Our test vector here is $\boldsymbol{y} = (1, 0, \ldots, 0)^T$, which gives us $\boldsymbol{y}^T\boldsymbol{M}(\alpha)\boldsymbol{y} = -\alpha < 1-\alpha$, so the same property holds for the star graph as the complete graph.

A.4 Proof of Proposition 2.4

(a) Let $\{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_N\}$ be an orthonormal basis of $\mathbb{R}^N$ with $\boldsymbol{f}_1 = \frac{1}{\sqrt{N}}\mathbf{1}_N$. It is easy to verify that these are eigenvectors of $\boldsymbol{L}$ [given in (2.27)] with corresponding eigenvalues $\lambda_1 = 0$ and $\lambda_k = \frac{N}{N-1}$ for $k = 2, \ldots, N$.

It follows from (2.46) that the diffusion process starting from $\boldsymbol{x}_0 = \boldsymbol{\delta}_{u_0}$ can be obtained as

$$\boldsymbol{x}(t) = \boldsymbol{f}_1 \boldsymbol{f}_1^T \boldsymbol{\delta}_{u_0} + e^{-\lambda_2 t}(\boldsymbol{I} - \boldsymbol{f}_1 \boldsymbol{f}_1^T)\boldsymbol{\delta}_{u_0}. \tag{A.15}$$

Assuming without loss of generality that $u_0 = 1$ and using the fact that $\boldsymbol{f}_1 = \frac{1}{\sqrt{N}}\mathbf{1}$, we have $\boldsymbol{x}(t) = \frac{1}{N}\left[1 + (N-1)e^{-\lambda_2 t}, 1 - e^{-\lambda_2 t}, \ldots, 1 - e^{-\lambda_2 t}\right]^T$. Using our knowledge of $\boldsymbol{L}$ and the fact that $\boldsymbol{P}_{u_0} = \mathrm{diag}(0, 1, \ldots, 1)$, it is now straightforward to compute the spreads as $\Delta_s^2(\boldsymbol{x}(t)) = \dfrac{Ne^{-2\lambda_2 t}}{1 + (N-1)e^{-2\lambda_2 t}}$ and $\Delta_{g,u_0}^2(\boldsymbol{x}(t)) = \dfrac{\frac{N-1}{N}\left(1 - e^{-\lambda_2 t}\right)^2}{1 + (N-1)e^{-2\lambda_2 t}}$. We can verify that these spreads satisfy (2.32). Thus, for all $t \geq 0$, $\boldsymbol{x}(t)$ achieves the uncertainty curve. $\Delta_s^2(\boldsymbol{x}(t))$ is continuous and $\lim_{t\to\infty} \Delta_s^2(\boldsymbol{x}(t)) = 0$, so $\eta_{u_0}(s) = \gamma_{u_0}(s)$ for $s \in (0, 1]$.

(b) Here, we assume without loss of generality that the center of the star, *i.e.*, the vertex with degree $N-1$ is $u_0 = 1$. Again, we explicitly construct an orthonormal eigenbasis for $\boldsymbol{L}$, given in this case by (2.33). In what follows, we will assume that $N > 2$; the star graph with 2 vertices is the same as the complete graph with 2 vertices, so the proof from (a) will apply to that case. Let $\boldsymbol{f}_1 = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2(N-1)}}, \ldots, \frac{1}{\sqrt{2(N-1)}}\right]^T$, $\boldsymbol{f}_N = \left[-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2(N-1)}}, \ldots, \frac{1}{\sqrt{2(N-1)}}\right]^T$, and $\boldsymbol{f}_k = \left[0, \boldsymbol{g}_k^T\right]^T$ for $k = 2, \ldots, N-1$, where $\{\boldsymbol{g}_k\}_{k=1}^{N-1}$ is any orthonormal basis for $\mathbb{R}^{N-1}$ satisfying $\boldsymbol{g}_1 = \frac{1}{\sqrt{N-1}}\mathbf{1}_{N-1}$. It is easy to verify that $\{\boldsymbol{f}_k\}_{k=1}^{N}$ forms an orthonormal basis for $\mathbb{R}^N$, and that the $\boldsymbol{f}_k$ are eigenvectors of $\boldsymbol{L}$ with corresponding eigenvalues $\lambda_1 = 0$, $\lambda_2 = \cdots = \lambda_{N-1} = 1$, and $\lambda_N = 2$.

Similar to (A.15), we can compute the diffusion process explicitly as

$$\begin{aligned}\boldsymbol{x}(t) &= \boldsymbol{f}_1\boldsymbol{f}_1^T\boldsymbol{\delta}_{u_0} + e^{-t}(\boldsymbol{I} - \boldsymbol{f}_1\boldsymbol{f}_1^T - \boldsymbol{f}_N\boldsymbol{f}_N^T)\boldsymbol{\delta}_{u_0} \qquad \text{(A.16)}\\ &\quad + e^{-2t}\boldsymbol{f}_N\boldsymbol{f}_N^T\boldsymbol{\delta}_{u_0}\\ &= \left(1 - e^{-t}\right)\boldsymbol{f}_1\boldsymbol{f}_1^T\boldsymbol{\delta}_{u_0} + \left(e^{-2t} - e^{-t}\right)\boldsymbol{f}_N\boldsymbol{f}_N^T\boldsymbol{\delta}_{u_0} \qquad \text{(A.17)}\\ &\quad + e^{-t}\boldsymbol{\delta}_{u_0}.\end{aligned}$$

Using the expressions for $\boldsymbol{f}_1$ and $\boldsymbol{f}_N$, we find that

$$\boldsymbol{x}(t) = \left[\frac{1}{2} + \frac{1}{2}e^{-2t}, \frac{1}{2\sqrt{N-1}}\left(1 - e^{-2t}\right)\mathbf{1}_{N-1}^T\right]^T.$$

From this, we can compute the graph spread as $\Delta_{g,u_0}^2(\boldsymbol{x}(t)) = \dfrac{(1-e^{-2t})^2}{2(1+e^{-4t})}$ and the spectral spread as $\Delta_s^2(\boldsymbol{x}(t)) = \dfrac{2e^{-4t}}{1+e^{-4t}}$. It is easy to verify that these spreads satisfy (2.34), and so $\boldsymbol{x}(t)$ achieves the uncertainty curve for $t \geq 0$. Once again, $\Delta_s^2(\boldsymbol{x}(t))$ is continuous and $\lim_{t\to\infty}\Delta_s^2(\boldsymbol{x}(t)) = 0$, so $\eta_{u_0}(s) = \gamma_{u_0}(s)$ for $s \in (0, 1]$.

A.5 Proof of Proposition 2.5

We know from Theorem 2.1 that every point on the uncertainty curve is achieved by an eigenvector associated with the smallest eigenvalue $q(\alpha)$ of $\boldsymbol{M}(\alpha) = \boldsymbol{P}_{u_0}^2 - \alpha\boldsymbol{L}$. In particular, the point $(1, 0)$ is achieved by $\boldsymbol{\delta}_{u_0}$, which is the eigenvector associated with the matrix $\boldsymbol{M}(0) = \boldsymbol{P}_{u_0}^2$ and eigenvalue $q(0) = 0$. Since $d(u_0, v) = 0$ if and only if $u_0 = v$ and $d(v, u_0) > 0$ otherwise, the eigenspace $\mathcal{S}(0)$ is one-dimensional. Thus, from the proof of Lemma 2.1 in Appendix A.2, there is some neighborhood $\mathcal{N}$ of

$\alpha = 0$ on which $\mathcal{S}(\alpha)$ is one-dimensional, and therefore $q(\alpha)$ is analytic. In this case, there exists some neighborhood of $s = 1$ for which we can use the parametric form of the uncertainty curve given in (2.25), namely $(s, \gamma_{u_0}(s)) = (s_u(\alpha), g_u(\alpha))$ where $s_u(\alpha) = -q'(\alpha)$ and $g_u(\alpha) = q(\alpha) - \alpha q'(\alpha)$ for $\alpha \in \mathcal{N}$.

We can thus compute the derivative of the uncertainty curve parametrically as

$$\frac{d\gamma_{u_0}}{ds} = \frac{g_u'(\alpha)}{s_u'(\alpha)} = \frac{-\alpha q''(\alpha)}{-q''(\alpha)} = \alpha. \tag{A.18}$$

where α is chosen so that $s(\alpha)$ is the argument at which we wish to evaluate the derivative. Similarly, the second derivative is

$$\frac{d^2\gamma_{u_0}}{ds^2} = \frac{\frac{d}{d\alpha}\left(\frac{g_u'(\alpha)}{s_u'(\alpha)}\right)}{s_u'(\alpha)} = \frac{1}{-q''(\alpha)}. \tag{A.19}$$

Both (A.18) and (A.19) require that $q''(\alpha)$ be nonzero. In what follows, we will explicitly compute $q''(0)$ and show that $q''(\alpha) \neq 0$ for $\alpha \in \mathcal{N}'$, where $\mathcal{N}' \subseteq \mathcal{N}$. As described in the proof of Lemma 2.1, there is an analytic eigenvector function $\boldsymbol{v}(\alpha)$ defined in a neighborhood of $\alpha = 0$ such that

$$\boldsymbol{M}(\alpha)\boldsymbol{v}(\alpha) = q(\alpha)\boldsymbol{v}(\alpha), \tag{A.20}$$

with $\boldsymbol{v}(0) = \boldsymbol{\delta}_{u_0}$ and $||\boldsymbol{v}(\alpha)||^2 = 1$. The spectral spread function is given by $s_u(\alpha) = \boldsymbol{v}(\alpha)\boldsymbol{L}\boldsymbol{v}(\alpha) = -q'(\alpha)$, where the second equality is due to (2.25). So we can compute

$$q''(\alpha) = -2\boldsymbol{v}(\alpha)\boldsymbol{L}\boldsymbol{v}'(\alpha). \tag{A.21}$$

To compute $\boldsymbol{v}'(\alpha)$, we differentiate both sides of (A.20) and after rearranging terms

obtain

$$[\boldsymbol{M}(\alpha) - q(\alpha)\boldsymbol{I}]\boldsymbol{v}'(\alpha) = \boldsymbol{L}\boldsymbol{v}(\alpha) + q'(\alpha)\boldsymbol{v}(\alpha). \tag{A.22}$$

From (A.20) and the fact that $\mathcal{S}(\alpha)$ is one-dimensional on $\mathcal{N}$, $\boldsymbol{M}(\alpha) - q(\alpha)\boldsymbol{I}$ has a one-dimensional nullspace spanned by $\boldsymbol{v}(\alpha)$. Since $0 = \frac{d}{d\alpha}||v(\alpha)||^2 = 2\boldsymbol{v}(\alpha)^T\boldsymbol{v}'(\alpha)$, when we multiply both sides of (A.22) by the Moore-Penrose pseudoinverse of $\boldsymbol{M}(\alpha) - q(\alpha)\boldsymbol{I}$, we obtain

$$\boldsymbol{v}'(\alpha) = [\boldsymbol{M}(\alpha) - q(\alpha)\boldsymbol{I}]^+\boldsymbol{L}\boldsymbol{v}(\alpha), \tag{A.23}$$

where we have also used the fact that $[\boldsymbol{M} - q(\alpha)\boldsymbol{I}]^+\boldsymbol{v}(\alpha) = 0$ to simplify the right-hand side of (A.23).

Setting $\alpha = 0$ and using the fact that $q(0) = 0$ and $\boldsymbol{v}(0) = \boldsymbol{\delta}_{u_0}$, we have $\boldsymbol{v}'(0) = \left(\boldsymbol{P}_{u_0}^2\right)^+\boldsymbol{L}\boldsymbol{\delta}_{u_0}$. Substituting this into (A.21), we get $q''(0) = -2\boldsymbol{\delta}_{u_0}^T\boldsymbol{L}\left(\boldsymbol{P}_{u_0}^2\right)^+\boldsymbol{L}\boldsymbol{\delta}_{u_0} = -2\sum_{v\sim u_0}\frac{(\boldsymbol{L}\boldsymbol{\delta}_{u_0})_v^2}{d^2(v,u_0)}$, where $(\boldsymbol{L}\boldsymbol{\delta}_{u_0})_v$ is the vth entry of $\boldsymbol{L}\boldsymbol{\delta}_{u_0}$. From the definition of the graph Laplacian, we have that for every $v \sim u_0$, $(\boldsymbol{L}\boldsymbol{\delta}_{u_0})_v = \frac{-1}{\sqrt{\deg u_0}}\frac{1}{\sqrt{\deg v}}$. Thus,

$$q''(0) = \frac{-2}{\deg u_0}\left(\sum_{v\sim u_0}\frac{1}{d(v,u_0)^2\deg v}\right). \tag{A.24}$$

Since the graph is connected, $q''(0) \neq 0$, and since $q(\alpha)$ is analytic on $\mathcal{N}$, there exists a neighborhood $\mathcal{N}' \subseteq \mathcal{N}$ containing 0 on which $q''(\alpha) \neq 0$ as well. Thus our expressions for the first and second derivatives (A.18) and (A.19) are valid at $s = 1$, which corresponds to $\alpha = 0$. We obtain $\left.\frac{d\gamma_{u_0}}{ds}\right|_{s=1} = 0$ and the expression for $\left.\frac{d^2\gamma_{u_0}}{ds^2}\right|_{s=1}$ given in (2.48).

To compute the derivatives of the curve $\eta_{u_0}(s)$ traced out by the diffusion process $\boldsymbol{x}(t)$, we express it parametrically in terms of t, with $(s, \eta_{u_0}(s)) = (s_d(t), g_d(t))$ where $s_d(t) = \frac{\boldsymbol{x}(t)^T \boldsymbol{L}\boldsymbol{x}(t)}{||\boldsymbol{x}(t)||^2}$ and $g_d(t) = \frac{\boldsymbol{x}(t)^T \boldsymbol{P}_{u_0}^2 \boldsymbol{x}(t)}{||\boldsymbol{x}(t)||^2}$.

We first show that $\dot{s}_d(t) < 0$. To simplify the computation of this and other derivatives, we introduce the function $R_{\boldsymbol{Z}}(t) = \frac{\boldsymbol{x}(t)^T \boldsymbol{Z}\boldsymbol{x}(t)}{||\boldsymbol{x}(t)||^2}$ for any fixed matrix $\boldsymbol{Z}$. It is easy to verify that since $\dot{\boldsymbol{x}}(t) = -\boldsymbol{L}\boldsymbol{x}(t)$, $\dot{R}_{\boldsymbol{Z}}(t) = 2R_{\boldsymbol{Z}}(t)R_{\boldsymbol{L}}(t) - R_{\boldsymbol{LZ}}(t) - R_{\boldsymbol{ZL}}(t)$, where the last two terms in the sum are equal if $\boldsymbol{Z}$ is symmetric. Since we have an explicit solution $\boldsymbol{x}(t) = e^{-\boldsymbol{L}t}\boldsymbol{\delta}_{u_0}$, we can see that $||\boldsymbol{x}(t)|| \neq 0$ for all t, so that $R_{\boldsymbol{Z}}(t)$ and its derivative is well-defined.

Since $s_d(t) = R_{\boldsymbol{L}}(t)$, we have

$$\dot{s}_d(t) = 2(s_d(t)^2 - R_{\boldsymbol{L}^2}(t)) = 2\left[(\boldsymbol{x}(t)^T \boldsymbol{L}\boldsymbol{x}(t))^2 - \boldsymbol{x}(t)^T \boldsymbol{L}^2 \boldsymbol{x}(t)\right] < 0$$

by the Cauchy-Schwarz inequality. Equality would hold only if $\boldsymbol{L}\boldsymbol{x}(t)$ were a multiple of $\boldsymbol{x}(t)$—*i.e.*, if $\boldsymbol{x}(t)$ were an eigenvector. From (2.46) we can see that this could only occur if $\boldsymbol{\delta}_{u_0}$ itself were an eigenvector, which is impossible for a connected graph. We can directly evaluate $s_d(0) = 1$ and $\lim_{t\to\infty} s_d(t) = 0$; combining this with the above result guarantees that $s_d(t)$ is a one-to-one function with range $(0, 1]$. Thus $\eta_{u_0}(s)$ is well-defined on that domain.

Since $g_d(t) = R_{\boldsymbol{P}_{u_0}^2}(t)$, we can compute the derivative

$$\dot{g}_d(0) = g_d(0)s_d(0) - R_{\boldsymbol{L}\boldsymbol{P}_{u_0}^2}(0) = 0.$$

Thus the diffusion curve's derivative at $s = 1$ is given by

$$\left.\frac{d\eta_{u_0}}{ds}\right|_{s=1} = \frac{\dot{g}_d(t)}{\dot{s}_d(t)} = 0 = \left.\frac{d\gamma_{u_0}}{ds}\right|_{s=1}. \tag{A.25}$$

Meanwhile, we can simplify the second derivative evaluated at $s = 1$, obtaining

$$\left.\frac{d^2\eta_{u_0}}{ds^2}\right|_{s=1} = \frac{\ddot{g}_d(0)}{\dot{s}_d^2(0)}. \tag{A.26}$$

The first derivative of $s_d(t)$ at $t = 0$ can be computed as

$$\begin{aligned}\dot{s}_d(0) &= 2(s_d(0)^2 - R_{\boldsymbol{L}^2}(0)) \\ &= 2\left(1 - ||\boldsymbol{L}\boldsymbol{\delta}_{u_0}||^2\right) = -2\sum_{v\sim u_0}\frac{1}{\deg u_0 \deg v}.\end{aligned} \tag{A.27}$$

The second derivative of $g_d(t)$ is

$$\begin{aligned}\ddot{g}_d(t) &= 2\dot{g}_d(t)s_d(t) + 2g_d(t)\dot{s}_d(t) - 4s_d(t)R_{\boldsymbol{L}\boldsymbol{P}_{u_0}^2}(t) \\ &\quad + 2R_{\boldsymbol{L}^2\boldsymbol{P}_{u_0}^2}(t) + 2R_{\boldsymbol{L}\boldsymbol{P}_{u_0}^2\boldsymbol{L}}(t)\end{aligned} \tag{A.28}$$

At $t = 0$, the only nonzero term in (A.28) is the last one:

$$\begin{aligned}\ddot{g}_d(0) &= 2R_{\boldsymbol{L}\boldsymbol{P}_{u_0}^2\boldsymbol{L}}(0) = 2\boldsymbol{\delta}_{u_0}^T\boldsymbol{L}\boldsymbol{P}_{u_0}^2\boldsymbol{L}\boldsymbol{\delta}_{u_0} \\ &= 2\sum_{v\sim u_0}\frac{d(v,u_0)^2}{\deg u_0 \deg v}.\end{aligned} \tag{A.29}$$

Now we can combine (A.26), (A.27), and (A.29) to obtain the expression for $\left.\frac{d^2\eta_{u_0}}{ds^2}\right|_{s=1}$ given in (2.48). By the Cauchy-Schwartz inequality,

$$\left(\sum_{v\sim u_0}\frac{1}{\deg v}\right)^2 \leq \left(\sum_{v\sim u_0}\frac{d(v,u_0)^2}{\deg v}\right)\left(\sum_{v\sim u_0}\frac{1}{d(v,u_0)^2\deg v}\right) \tag{A.30}$$

with equality *if and only if* $d(v,u_0)^2 = c$ for every $v \sim u_0$, where c is some constant. Comparing the expressions for the second derivatives of the uncertainty curve and diffusion curve, we can see that $\frac{d^2\eta_{u_0}}{ds^2}\Big|_{s=1} \geq \frac{d^2\gamma_{u_0}}{ds^2}\Big|_{s=1}$, with equality *if and only if* $d(v,u_0)$ is identical for every $v \sim u_0$.

Appendix B

Random Walk Detection Proofs

B.1 Proof of Lemma 3.1

The proof is a rather straightforward generalization of the proof of the standard Chernoff-Stein lemma given in [41]. Consider the sequence of optimal detectors δ_N, *i.e.*, the Neyman-Pearson detector that choose $\mathcal{H}_1$ if $\ell_N > \tau_N$ and $\mathcal{H}_0$ otherwise, where τ_N is a sequence of thresholds chosen to satisfy the false alarm constraint $P_{\text{false\_alarm}} \leq \epsilon$ for some fixed $\epsilon \in (0, 1)$. The false alarm and miss detection probabilities are then given by

$$P^N_{\text{false\_alarm}} = P_0(\ell_N > \tau_N)$$

and

$$P^N_{\text{miss}} = P_1(\ell_N < \tau_N),$$

respectively. Note that $\liminf_{N\to\infty} \tau_N \geq -\kappa$ must hold; if that were not the case, then since $\ell_N \to -\kappa$ in probability under $\mathcal{H}_0$, we would have $\limsup_{N\to\infty} P^N_{\text{false\_alarm}} = 1$, which would violate the false alarm constraint.

Noting that $P_1(\boldsymbol{Y}^N) = \exp(N\ell_N)P_0(\boldsymbol{Y}^N)$, we can rewrite the miss detection probability as

$$\begin{aligned} P^N_{\text{miss}} &= \mathbb{E}_1\, \mathbb{1}(\ell_N < \tau_N) \\ &= \mathbb{E}_0\, \mathbb{1}(\ell_N < \tau_N) \exp(N\ell_N), \end{aligned} \tag{B.1}$$

where $\mathbb{1}(\cdot)$ is the indicator function, since multiplying by $\exp(N\ell_N)$ converts the density $P_0(\cdot)$ to the density $P_1(\cdot)$. Choosing an arbitrary $\delta > 0$, we have

$$\begin{aligned} &P_0(\ell_N \in [-\kappa - \delta, \tau_N]) \\ &= 1 - P_0(\ell_N < -\kappa - \delta) - P_0(\ell_N > \tau_N) &\text{(B.2)} \\ &\geq 1 - P_0(\ell_N < -\kappa - \delta) - \epsilon, &\text{(B.3)} \end{aligned}$$

since the final term in (B.2) is just the false alarm probability, which is constrained. It then holds that

$$\begin{aligned} \frac{1}{N}\log P^N_{\text{miss}} &= \frac{1}{N}\log \mathbb{E}_0\big[\mathbb{1}(\ell_N < \tau_N)\exp(N\ell_N)\big] \\ &\geq \frac{1}{N}\log \mathbb{E}_0\big[\mathbb{1}(\ell_N \in [-\kappa-\delta, \tau_N])\exp(N\ell_N)\big] \\ &\geq -\kappa - \delta + \frac{1}{N}\log P_0\big(\ell \in [-\kappa-\delta, \tau_N]\big) \\ &\geq -\kappa - \delta + \frac{1}{N}\log\left[1 - \epsilon - \underbrace{P_0(\ell_N < -\kappa - \delta)}_{\longrightarrow 0}\right], &\text{(B.4)} \end{aligned}$$

from which we can conclude that $\liminf_{N\to\infty} \frac{1}{N} \log P^N_{\text{miss}} \geq -\kappa$, since δ can be made arbitrarily small and the last term on the right-hand side of (B.4) vanishes.

Now, instead, suppose we simply fix $\tau_N = -\kappa + \delta$ for every N. Then clearly $P^N_{\text{false\_alarm}} \to 0$ because $\ell_N \to -\kappa$ in probability. Thus, eventually, $P^N_{\text{false\_alarm}} < \epsilon$. Meanwhile, the maximum value of the quantity inside the integral in (B.1) is $\exp(N\tau_N)$, so we have that $\frac{1}{N} \log P^N_{\text{miss}} \leq \tau_N = -\kappa + \delta$. So $\limsup_{N\to\infty} \frac{1}{N} \log P^N_{\text{miss}} \leq -\kappa$, since again δ is arbitrary.

We have shown the following: (1) any sequence of Neyman-Pearson detectors satisfying the false alarm constraint $P^N_{\text{false\_alarm}} < \epsilon$ satisfies

$$\liminf_{N\to\infty} \frac{1}{N} \log P^N_{\text{miss}} \geq -\kappa,$$

and (2) there exists a sequence of Neyman-Pearson detectors satisfying the false alarm constraint $P_{\text{false\_alarm}} < \epsilon$ for which $\limsup_{N\to\infty} \frac{1}{N} \log P^N_{\text{miss}} \leq -\kappa$. Thus for the optimal sequence of detectors, we have $\eta \stackrel{\text{def}}{=} \lim_{N\to\infty} -\frac{1}{N} \log P^N_{\text{miss}} = \kappa$. This holds for any $\epsilon \in (0, 1)$, so the proposition is proved.

B.2 Proof of Proposition 3.3 [Properties of $\log \lambda_t$]

(1) $\boldsymbol{P}^{(t)}$ is an irreducible nonnegative matrix for any t, just as $\boldsymbol{P}$ is. Thus the Perron-Frobenius theorem tells us that $\lambda_{\max}(\boldsymbol{P}^{(t)})$ is a real, positive eigenvalue, so $\log \lambda_t$ is well-defined and finite. Since w^t is an analytic function of t for any positive w and the zero function is an analytic function, we have that every entry of $\boldsymbol{P}^{(t)}$ is analytic

in t. Standard perturbation-theoretic results [73] tell us that on any neighborhood in which a matrix function is analytic and an eigenvalue remains isolated from the rest of the spectrum (*i.e.* has no multiplicity), it can be analytically continued to the rest of that neighborhood. Since λ_t is the Perron-Frobenius eigenvalue, it is simple and thus isolated. Therefore, it is an analytic function of t everywhere. Since it is positive, $\log \lambda_t$ is analytic as well. The convexity of $\log \lambda_t$ follows from a property of Hadamard powers proven by Horn and Johnson in [66, p. 361]: for any nonnegative matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ and $0 \leq \alpha \leq 1$, they showed that

$$\lambda_{\max}(\boldsymbol{A}^{(\alpha)} \circ \boldsymbol{B}^{(1-\alpha)}) \leq \lambda_{\max}(\boldsymbol{A})^{\alpha} \lambda_{\max}(\boldsymbol{B})^{1-\alpha}. \tag{B.5}$$

Taking $\boldsymbol{A} = \boldsymbol{P}^{(s)}$ and $\boldsymbol{B} = \boldsymbol{P}^{(t)}$ for arbitrary $t > s > 0$, and using the fact that $\log$ is an increasing function, we have

$$\log \lambda_{\max}(\boldsymbol{P}^{(\alpha s+(1-\alpha)t)}) \leq \alpha \log \lambda_{\max}(\boldsymbol{P}^{(s)}) + (1-\alpha) \log \lambda_{\max}(\boldsymbol{P}^{(t)}), \tag{B.6}$$

which by definition gives us the convexity of $\log \lambda_t$.

(2) Strict convexity means that the inequality in (B.6) must be strict. Since $\log$ is strictly increasing, equality holds if and only if it holds in (B.5), which for irreducible matrices holds if and only if there exists a positive scalar γ and a positive diagonal matrix $\boldsymbol{D}$ such that $\gamma \boldsymbol{A} = \boldsymbol{D}^{-1} \boldsymbol{B} \boldsymbol{D}$ [66, p. 361]. For our problem, then, equality holds if and only if there are some $t > s > 0$ such that for all i, j

$$\gamma p_{ij}^{s} = \frac{d_j}{d_i} p_{ij}^{t},$$

for some positive constants γ and $d_i, i = 1, \ldots, M$. Thus either $p_{ij} = 0$ or

$$p_{ij} = \gamma^{\frac{1}{t-s}} d_i^{\frac{1}{t-s}} d_j^{\frac{1}{s-t}}.$$

Summing over j on both sides of this equation tells us that d_i must be a constant. This means that all of the nonzero entries of $\boldsymbol{P}$ must be constant. Since the row sums of $\boldsymbol{P}$ must be one, this means that every row of $\boldsymbol{P}$ having exactly K nonzero entries equal to $\frac{1}{K}$ for some $K \leq M$ is the only situation in which strict convexity does not hold.

So what exactly is $\log \lambda_t$ in that case? Consider the test vector $\mathbf{1}$: we have $\boldsymbol{P}^{(t)}\mathbf{1} = K^{1-t}\mathbf{1}$. So the test vector is an eigenvector. The Perron-Frobenius theorem states that any positive eigenvector must correspond to the largest eigenvalue. Since $\mathbf{1}$ has all positive entries, we have that $\lambda_t = K^{1-t}$, so $\log \lambda_t = (1-t) \log K$.

(3) As before, we use the perturbation results. In addition to an analytic eigenvalue function, in the case of a simple eigenvalue, there are analytic functions for the left- and right-eigenvectors. These can be normalized as desired. So we have analytic functions $\boldsymbol{a}_t$ and $\boldsymbol{b}_t$ such that

$$\boldsymbol{a}_t^T \boldsymbol{P}^{(t)} = \lambda_t \boldsymbol{a}_t^T$$

$$\boldsymbol{P}^{(t)} \boldsymbol{b}_t = \lambda_t \boldsymbol{b}_t$$

and normalized[1] such that $\boldsymbol{a}_t^T \boldsymbol{b}_t = 1$ and $\boldsymbol{a}_t^T \mathbf{1} = 1$. We can write the largest eigenvalue function as $\lambda_t = \boldsymbol{a}_t^T \boldsymbol{P}^{(t)} \boldsymbol{b}_t$. Using the chain rule, we can compute the deriva-

[1] They are Perron-Frobenius eigenvectors, so they are nonnegative can never be orthogonal.

tive

$$\begin{aligned}\lambda_t' &= (\boldsymbol{a}_t')^T \boldsymbol{P}^{(t)} \boldsymbol{b}_t + \boldsymbol{a}_t^T \boldsymbol{P}^{(t)} \boldsymbol{b}_t' + \boldsymbol{a}_t^T ((\log \boldsymbol{P}) \circ \boldsymbol{P}^{(t)}) \boldsymbol{b}_t \\ &= \lambda_t [(\boldsymbol{a}_t')^T \boldsymbol{b}_t + \boldsymbol{a}_t^T \boldsymbol{b}_t'] + \boldsymbol{a}_t^T ((\log \boldsymbol{P}) \circ \boldsymbol{P}^{(t)}) \boldsymbol{b}_t \\ &= \boldsymbol{a}_t^T [(\log \boldsymbol{P}) \circ \boldsymbol{P}^{(t)}] \boldsymbol{b}_t, \end{aligned} \tag{B.7}$$

where in reaching (B.7) we have used the fact that $\boldsymbol{a}_t^T \boldsymbol{b}_t = 1$ and thus $(\boldsymbol{a}_t')^T \boldsymbol{b}_t + \boldsymbol{a}_t^T \boldsymbol{b}_t' = 0$. Using the chain rule, we have that $\frac{d}{dt} \log \lambda_t = \frac{\lambda_t'}{\lambda_t}$, and the result follows. Note that the normalization of the eigenvectors is irrelevant in the final expression because the normalization factors will cancel out in the numerator and denominator.

(4) Since $\log \lambda_t$ is convex, its derivative is nondecreasing. Thus

$$\begin{aligned}\inf_t \frac{d}{dt} \log \lambda_t &= \lim_{t \to -\infty} \frac{d}{dt} \log \lambda_t \\ &= \lim_{t \to -\infty} \frac{\log \lambda_t}{t}, \end{aligned}$$

where the finals step results from L'Hôpital's rule. By the same argument, we have

$$\sup_t \frac{d}{dt} \log \lambda_t = \lim_{t \to +\infty} \frac{\log \lambda_t}{t}$$

Horn and Johnson show that these are equal to $\rho_{\min}$ and $\rho_{\max}$, respectively [66, p. 367].

B.3 Proof of Proposition 3.5 [Properties of $s(\rho)$]

As we stated in Proposition 3.3, if $\boldsymbol{P}$ is the transition matrix for a uniform random walk on a regular graph, then $\log \lambda_t = (1-t)\log K$. If $\rho = -\log K$, then $\log \lambda_t - t\rho = \log K$, a constant, giving us $s(-\log K) = \log K$. For any other ρ, the function $\log \lambda_t - t\rho$ is linear but not constant, so it is unbounded and has an infimum of $-\infty$. Now consider the general case:

(1) Consider the function $-s(\rho) = \sup_t \{t\rho - \log \lambda_t\}$. This is the convex conjugate of $\log \lambda_t$. A convex conjugate function is guaranteed to be a convex function with range $\mathbb{R} \bigcup \{+\infty\}$, so $s(\rho)$ is a concave function with range $\mathbb{R} \bigcup \{-\infty\}$. Since $\log \lambda_t$ is strictly convex, the infimum $\inf_t \log \lambda_t - t\rho$ that defines $s(\rho)$ is achieved at no more than one point t^* [110]. Since it is also differentiable, then if and only if the infimum is achieved at t^*, we have

$$\frac{d}{dt} \log \lambda_t \Big|_{t^*} = \rho.$$

If there is no such t^*, then $s(\rho) = -\infty$. This will be the case if $\rho < \rho_{\min}$ or $\rho > \rho_{\max}$. Suppose, however, that $\rho \in (\rho_{\min}, \rho_{\max})$. By the intermediate value theorem, there must be some t^* for which $\frac{d}{dt} \log \lambda_t \Big|_{t^*} = \rho$. Then $s(\rho) = \log \lambda_{t^*} - t^*\rho$. It remains to prove nonnegativity.

We use the following alternate expressions [66, p. 367] for $\rho_{\min}$ and $\rho_{\max}$:

$$\rho_{\min} = \min_{\substack{\text{self-avoiding loops} \\ i_1,\ldots,i_L \\ 1 \le L \le M}} \frac{1}{L} \sum_{j=1}^{L} \log p_{i_j,i_{j+1}} \tag{B.8}$$

$$\rho_{\max} = \max_{\substack{\text{self-avoiding loops} \\ i_1,\ldots,i_L \\ 1 \le L \le M}} \frac{1}{L} \sum_{j=1}^{L} \log p_{i_j,i_{j+1}}, \tag{B.9}$$

where the suprema are over self-avoiding loops that obey the topology induced by the sparsity of $\boldsymbol{P}$, so each $i_1, \ldots, i_L$ is unique, $p_{i_j,i_{j+1}} \neq 0$, and we use the convention that $i_{L+1} = i_1$.) Let $i_1^*, \ldots, i_L^*$ be the self-avoiding loop achieving the maximum in (B.9). Define the matrix $\boldsymbol{B}$ as follows: every transition in the maximal loop is given the same value as in $\boldsymbol{P}^{(t)}$ (*i.e.*, $[\boldsymbol{B}]_{i_1^*,i_2^*} = p^t_{i_1^*,i_2^*}, \ldots, [\boldsymbol{B}]_{i_L^*,i_1^*} = p^t_{i_L^*,i_1^*}$), and every other entry is set to 0. On an elementwise basis, then, $\boldsymbol{P}^{(t)} \geq \boldsymbol{B}$. If $L < M$, then $\boldsymbol{B}$ is not irreducible, but the Perron-Frobenius theorem still guarantees us that it has a real eigenvalue $\lambda_{\max}(\boldsymbol{B})$ equal to its spectral radius [67]. It is not hard to verify that taking powers of $\boldsymbol{B}$ eventually results in a constant multiple of a diagonal $0-1$ matrix:

$$\begin{aligned} \boldsymbol{B}^L &= p^t_{i_1^*,i_2^*} \cdots p^t_{i_L^*,i^*+1} \boldsymbol{D} \\ &= \exp(tL\rho_{\max}) \boldsymbol{D}. \end{aligned} \tag{B.10}$$

Here, the diagonal entries of $\boldsymbol{D}$ associated with the indices $i_1^*, \ldots, i_L^*$ are 1, and the others are all 0. Now if we let $\boldsymbol{v}$ be an eigenvector of $\boldsymbol{B}$ associated with the eigenvalue $\lambda_{\max}(\boldsymbol{B})$ whose only nonzero entries are those associated with the indices

$i_1^*, \ldots, i_L^*$, we have

$$\boldsymbol{B}^L \boldsymbol{v} = \lambda_{\max}(\boldsymbol{B})^L \boldsymbol{v}. \tag{B.11}$$

Combining this with (B.10), we obtain

$$\lambda_{\max}(\boldsymbol{B}) = \exp(t\rho_{\max})$$

Since the Perron-Frobenius eigenvalue $\lambda_{\max}(\cdot)$ is a monotonic function of the matrix entries [66], we have $\log \lambda_t \geq \log \lambda_{\max}(\boldsymbol{B}^{(t)}) = t\rho_{\max}$ for every t. Now since $\log \lambda_t - t\rho_{\max} \geq 0$ for all t, we must have $s(\rho_{\max}) \geq 0$. A similar argument shows that $s(\rho_{\min}) \geq 0$ as well. It then follows from the concavity of $s(\rho)$ that $s(\rho) \geq 0$ on $[\rho_{\min}, \rho_{\max}]$.

(2) Any proper convex function is continuous on the interior of its effective domain, so $-s(\rho)$ is continuous on $(\rho_{\min}, \rho_{\max})$, and thus $s(\rho)$ is as well. Since $-s(\rho)$ is the Legendre-Fenchel transform of $\log \lambda_t$, which is itself a convex function, it must be lower semicontinuous. So $s(\rho)$ must be upper semicontinuous, and therefore it is continuous from above at $\rho_{\min}$ and $\rho_{\max}$.

(3) Since $\log \lambda_t$ is strictly convex (remember, we are not considering regular graphs here), there is at most one point that achieves the infimum $\inf_t \{\lambda_t - t\rho\}$ that defines $s(\rho)$. We showed earlier that as long as $\rho \in (\rho_{\min}, \rho_{\max})$, there is exactly one such point. Another basic result in convex analysis [110, Theorem 11.8] tells us that $s(\cdot)$ is then differentiable at ρ, and in particular $-s'(\rho) = t_\rho$, where t_ρ is the argument of

the minimum. Since $\log \lambda_t$ is differentiable, we also have that

$$\frac{d}{dt} \log \lambda_t \Big|_{t=t_\rho} = \rho.$$

Thus $-s'(\rho)$ is the inverse function of $\frac{d}{dt} \log \lambda_t$ as claimed. Since $\log \lambda_t$ is strictly convex, its derivative is one-to-one, and thus so is $s'(\rho)$.

(4) We know that $s(\rho) = \log \lambda_{t_\rho} - t_\rho \rho$, where t_ρ is the value of t at which $\frac{d}{dt} \log \lambda_t = \rho$, if such a value exists, and $s'(\rho) = -t_\rho$; otherwise $s(\rho) = -\infty$. Using the expression for the derivative from the proof of Proposition 3.3, and the fact that the left and right eigenvectors of $\boldsymbol{P}$ are $\boldsymbol{a}_1 = \boldsymbol{\pi}$ and $\boldsymbol{b}_1 = \mathbf{1}$, respectively, we have that

$$\begin{aligned} \frac{d}{dt} \lambda_t \Big|_{t=1} &= \frac{\boldsymbol{\pi}^T [(\log \boldsymbol{P}) \circ \boldsymbol{P}] \mathbf{1}}{\boldsymbol{\pi}^T \boldsymbol{P} \mathbf{1}} \\ &= \sum_i \pi_i \sum_j p_{ij} \log p_{ij} \\ &= -H. \end{aligned}$$

Meanwhile, $\lambda_1 = 1$, so $\frac{d}{dt} \log \lambda_t \Big|_{t=1} = -H$. So $s(-H) = \log 1 - 1 \cdot (-H) = H$, and $s'(-H) = -1$. The same argument gives us $s(\rho_0)$ and $s'(\rho_0)$, only without the nicely-interpretable values.

B.4 Proof of Theorem 3.2

To prove the statement of the theorem, we need to show that the upper bound

$$\limsup_{N \to \infty} \frac{1}{N} \log Q_N(B) \leq - \inf_{(\rho, \xi) \in B} I(\rho, \xi) \tag{B.12}$$

holds almost surely for every closed set $B \subset \mathbb{R}^2$, and the lower bound

$$\liminf_{N\to\infty} \frac{1}{N} \log Q_N(U) \geq - \inf_{(\rho,\xi)\in U} I(\rho,\xi) \tag{B.13}$$

holds almost surely for every open set $U \subset \mathbb{R}^2$. We will use an argument parallel to Dorlas and Wedagedera's for the random energy model with an external field [47]. Let $\mathcal{A} = \left\{(\rho,\xi) : I_1(\rho) + \frac{\xi^2}{2} \leq \log \lambda_0\right\}$ be the effective domain of $I(\cdot,\cdot)$, *i.e.* the set on which it is finite. It can also be written as $\mathcal{A} = \left\{(\rho,\xi) : |\xi| \leq \sqrt{2s(\rho)}\right\}$. It is the union of hypograph of the function $\xi = \sqrt{2s(\rho)}$ and its reflection over the ρ axis. We know from Proposition 3.5 that $s(\rho)$ is nonnegative and concave on $[\rho_{\min}, \rho_{\max}]$. Since $\sqrt{\cdot}$ is a concave and increasing function, we have that $\mathcal{A}$ is a convex set. A notional illustration of $\mathcal{A}$ was shown in Figure 3.3.

We will be able to build up the result for general sets by studying the behavior of a few classes of primitive sets. Consider a box $C = [\rho, \rho+\delta] \times [\xi, \xi+\delta]$ with sides of length δ; suppose first that it is entirely outside of $\mathcal{A}$. By definition, $Q_N(C) = \frac{1}{\#\mathcal{P}^N}\#\{\boldsymbol{s} : \frac{1}{N}\log P(\boldsymbol{s}) \in [\rho, \rho+\delta], \frac{1}{N}x_{\boldsymbol{s}} \in [\xi, \xi+\delta]\}$. So $\#\mathcal{P}^N Q_N(C)$ is a binomial random variable with parameters $\#\mathcal{P}^N Q_N^1([\rho, \rho+\delta])$ and $\sqrt{\frac{N}{2\pi}}\int_\xi^{\xi+\delta} \exp(-\frac{Nx^2}{2})dx$. This means that

$$\mathbb{E}Q_N(C) = Q_N^1([\rho, \rho+\delta])\sqrt{\tfrac{N}{2\pi}}\int_\xi^{\xi+\delta} \exp(-\tfrac{Nx^2}{2})dx \tag{B.14}$$

and

$$\mathrm{var}(Q_N(C)) = \frac{1}{\#\mathcal{P}^N} Q_N^1([\rho, \rho+\delta]) \left(\sqrt{\tfrac{N}{2\pi}} \int_\xi^{\xi+\delta} \exp(-\tfrac{Nx^2}{2}) dx \right) \cdot \left(1 - \sqrt{\tfrac{N}{2\pi}} \int_\xi^{\xi+\delta} \exp(-\tfrac{Nx^2}{2}) dx \right).$$

Now for any ϵ, we can choose N' large enough so that for every $N > N'$,

$$\begin{aligned}
&P(Q_N(C) > 0) \\
&\quad = P(\#\mathcal{P}^N Q_N(C) \geq 1) && \text{(B.15)} \\
&\quad \leq \mathbb{E}(\mathcal{P}^N Q_N(C)) && \text{(B.16)} \\
&\quad = \#\mathcal{P}^N Q_N^1([\rho, \rho+\delta]) \sqrt{\tfrac{N}{2\pi}} \int_\xi^{\xi+\delta} \exp(-\tfrac{Nx^2}{2}) dx && \text{(B.17)} \\
&\quad \leq \exp(N\left[\log \lambda_0 + \epsilon\right]) \exp\left(N[\epsilon - \inf_{r \in [\rho, \rho+\delta]} I_1(r)] \right) \delta \sqrt{\tfrac{N}{2\pi}} \exp\left(N \left[-\inf_{x \in [\xi, \xi+\delta]} \tfrac{x^2}{2} \right] \right) \\
&\quad = \exp\left(N \Big[-\inf_{\substack{r \in [\rho, \rho+\delta] \\ x \in [\xi, \xi+\delta]}} (I_1(r) + \tfrac{x^2}{2}) + \log \lambda_0 + 2\epsilon \Big] + \log \delta + \tfrac{1}{2} \log \tfrac{N}{2\pi} \right) && \text{(B.18)} \\
&\quad \longrightarrow 0.
\end{aligned}$$

Here, (B.15) is because there is a discrete number of paths, (B.16) is the Markov inequality, and (B.17) is due to (B.14). (B.18) converges to 0 because the coefficient on N in the exponent is guaranteed to be negative for small enough ϵ because C is entirely outside of $\mathcal{A}$. We have merely proven convergence in probability, but since the probability goes to zero exponentially fast, the Borel-Cantelli lemma tells us that with probability 1, there is an N' such that for every $N > N'$, we have $Q_N(C) = 0$, so $\lim_{N \to \infty} \frac{1}{N} \log Q_N(C) = -\infty$ almost surely.

If instead we consider the half-planes $C = \{(\rho, \xi) : \xi > \sqrt{2\log\lambda_0} + 1\}$ or $C = \{(\rho, \xi) : \xi < -\sqrt{2\log\lambda_0} - 1\}$, we can use the same argument (replacing the Gaussian integrals in (B.17) with standard Gaussian tail bounds, and using the fact that C is outside of $\mathcal{A}$) to show that $\lim_{N\to\infty} \frac{1}{N}\log Q_N(C) = -\infty$ almost surely for these sets as well. The half planes $C = \{(\rho, \xi) : \rho > \rho_{\max} + 1\}$ and $C = \{(\rho, \xi) : \rho < \rho_{\min} - 1\}$ also contain no states for large enough N due to the definitions of $\rho_{\min}$ and $\rho_{\max}$, so again $\lim_{N\to\infty} \frac{1}{N}\log Q_N(C) = -\infty$ almost surely.

Now, suppose we have a box $C = [\rho, \rho+\delta] \times [\xi, \xi+\delta]$, but this time it intersects the set $\mathcal{A}$. By Chebyshev's inequality we know that for any ϵ, there is an N' such that for $N > N'$,

$$\begin{aligned}
&P\Big(|Q_N(C) - \mathbb{E}Q_N(C)| \geq k\mathbb{E}Q_N(C)\Big) \\
&\qquad \leq \frac{1}{k^2}\frac{\mathrm{var}(Q_N(C))}{(\mathbb{E}Q_N(C))^2} \\
&\qquad \leq \frac{1}{k^2}\left(\#\mathcal{P}^N \cdot Q_N^1([\rho, \rho+\delta]) \cdot \sqrt{\tfrac{N}{2\pi}}\int_{\xi}^{\xi+\delta} \exp(-\tfrac{Nx^2}{2})dx\right)^{-1} \\
&\qquad \leq \frac{1}{k^2}\exp\Big(-N[\log\lambda_0 - \epsilon - \epsilon - \inf_{r\in[\rho,\rho+\delta]} I_1(r) - \inf_{x\in[\xi,\xi+\delta]} \tfrac{x^2}{2}] - \log\delta - \tfrac{1}{2}\log\left(\tfrac{N}{2\pi}\right)\Big).
\end{aligned}$$

By choosing ϵ small enough, we can guarantee that this probability decays exponentially in N. Thus, by the Borel-Cantelli lemma $\lim_{N\to\infty} \frac{Q_N(C)}{\mathbb{E}Q_N(C)} = 1$ with probability 1. Because $\log(\cdot)$ is continuous at 1, this gives us $\lim_{N\to\infty} \frac{1}{N}\log Q_N(C) = \lim_{N\to\infty} \frac{1}{N}\log \mathbb{E}Q_N(C)$. Using (B.14), we can compute

$$\lim_{N\to\infty} \frac{1}{N}\log Q_N(C) = -\inf_{(\rho,\xi)\in C}\Big\{I_0(r) + \frac{x^2}{2}\Big\} = -\inf_{(\rho,\xi)\in C} I(r, x),$$

almost surely.

Using these primitives, we can prove the large deviation property directly. We start with the upper bound. Suppose B is a closed set entirely outside of the effective domain of $I(\cdot,\cdot)$, *i.e.* $B \cap \mathcal{A} = \emptyset$. Let $d(B,\mathcal{A})$ be the distance between the set B and $\mathcal{A}$. Then if we choose some $\delta < d(B,\mathcal{A})/\sqrt{2}$, the set B can be covered by a finite number of $\delta \times \delta$ boxes that are entirely outside of $\mathcal{A}$ plus possibly one or more of the half-planes described above: $B \subset \bigcup_{\ell=1}^{L} B_\ell$, where each B_ℓ is one of the primitives described above and $B_\ell \cap \mathcal{A} = \emptyset$. Then we have

$$\begin{aligned}\limsup_{N\to\infty} \frac{1}{N}\log Q_N(B) &\le \limsup_{N\to\infty} \frac{1}{N}\log \sum_{\ell=1}^{L} Q_N(B_\ell) \\ &\le \lim_{N\to\infty}\frac{1}{N}\log L + \limsup_{N\to\infty}\frac{1}{N}\log(\max_\ell Q_N(B_\ell)) \\ &= -\infty, \end{aligned} \tag{B.19}$$

almost surely, since the maximum is over just a finite number of sets.

Next suppose B is a closed set that is not entirely outside the effective domain: $B \cap \mathcal{A} \neq \emptyset$. Let $b = \inf_{(\rho,\xi)\in B} I(\rho,\xi)$. Because I is continuous inside $\mathcal{A}$, for any ϵ, we can choose a δ and cover B with the primitive halfplanes and a finite number of boxes of width δ such that, for each square (and each halfplane, trivially) B_ℓ, we

have $\inf_{(\rho,\xi)\in B_\ell} I(\rho,\xi) \geq b - \epsilon$. We have

$$\begin{aligned}
\limsup_{N\to\infty} \frac{1}{N}\log Q_N(B) &\leq \limsup_{N\to\infty} \frac{1}{N}\log \sum_{\ell=1}^{L} Q_N(B_\ell) \\
&\leq \lim_{N\to\infty} \frac{1}{N}\log L + \limsup_{N\to\infty} \frac{1}{N}\log(\max_\ell Q_N(B_\ell)) \\
&= \max_\ell \left\{ \limsup_{N\to\infty} \frac{1}{N}\log(Q_N(B_\ell)) \right\} \\
&\leq -(b-\epsilon).
\end{aligned}$$

Since ϵ can be made arbitrarily small, we have

$$\limsup_{N\to\infty} \frac{1}{N}\log Q_N(B) \leq -\inf_{(\rho,\xi)\in B} I(\rho,\xi). \tag{B.20}$$

Combining this with (B.19) gives us the large deviation upper bound for any closed set B.

Now we must prove the large deviations lower bound (B.13). Let U be an open set. First, suppose $U \cap \mathcal{A} = \emptyset$, meaning that the set is entirely outside the region. Then the lower bound is trivial: it amounts to proving that

$$\liminf_{N\to\infty} \frac{1}{N}\log Q_N(U) \geq -\infty,$$

which is obviously true. So let us assume that $U \cap \mathcal{A} \neq \emptyset$. For any $\epsilon > 0$, there is a square C of width δ contained entirely within U such that $\inf_{(r,x)\in C} I(r,x) < \inf_{(r,x)\in U} I(r,x) + \epsilon$, by the following argument. First, note that the infimum must be achieved on the interior of $\mathcal{A}$[2]. If the infimum is achieved at a point (ρ^*, ξ^*)

[2]To see this, note that the boundary points of $\mathcal{A}$ satisfy $\frac{\xi^2}{2} - s(\rho) = 0$, in which case $I(\rho,\xi) = \log\lambda_0$ is the maximum possible value of I, or either $\rho = \rho_{\min}, |\xi| \leq s(\rho_{\min})$ or $\rho = \rho_{\max}, |\xi| \leq s(\rho_{\max})$, in which case the concavity of $s(\rho)$ tells us that we can decrease I by moving into the interior of $\mathcal{A}$.

on the interior of U, then we can easily just draw a box C around it that is small enough to fit in U, and it must have the same infimum. If on the other hand the infimum is achieved on the boundary of U, the continuity of $I(\rho, \xi)$ means that we can choose a small open neighborhood around (ρ^*, ξ^*) in which $I(\rho, \xi) < I(\rho^*, \xi^*) + \epsilon$. This neighborhood must intersect with U since it is centered on a boundary point, and the intersection must be an open set since both sets are open. Then we can choose a small box C that fits inside the intersection, and again we must have that $\inf_{(r,x)\in C} I(r,x) < \inf_{(r,x)\in U} I(r,x) + \epsilon$.

Using our result for boxes, we have

$$\begin{aligned}\liminf_{N\to\infty} \frac{1}{N} \log Q_N(U) &\geq \liminf_{N\to\infty} \frac{1}{N} \log Q_N(C) \\ &= -\inf_{(r,x)\in C} I(r,x) \\ &> -\inf_{(r,x)\in U} I(r,x) - \epsilon,\end{aligned}$$

almost surely. Since this holds for any ϵ, the lower bound is proved.

B.5 Uniform integrability of the free energy density

In this appendix, we show that the sequence of random variables

$$X_N \stackrel{\text{def}}{=} \frac{1}{N} \log\Big(\sum_{\boldsymbol{s}\in\mathcal{P}^N} P(\boldsymbol{s}) \exp(\beta x_{\boldsymbol{s}}) \Big), \qquad N = 1, 2, \ldots \tag{B.21}$$

is *uniformly integrable*. Our arguments will closely follow those in Olivieri and Picco [98], who showed that the free energy density of the standard random energy

model [42] is uniformly integrable. We start by recalling the definition of uniform integrability:

Definition B.1. *A sequence of random variables $\{X_N\}_{N\geq 1}$ is uniformly integrable if*

$$\lim_{\alpha\to\infty} \sup_{N>N_0} \mathbb{E}\left(\mathbb{1}_{|X_N|\geq\alpha}|X_N|\right) = 0, \tag{B.22}$$

for some $N_0 > 0$.

To proceed, we first note that, by the definition of $\rho_{\min}$ in (3.20), there exists some N_0 such that $P(\boldsymbol{s}) > \exp(2N\rho_{\min})$ for all $N \geq N_0$. (Note that $\rho_{\min}$ is negative, so $2\rho_{\min}$ is actually less than $\rho_{\min}$.) Using this inequality and the fact that $\sum_{\boldsymbol{s}\in\mathcal{P}^N} P(\boldsymbol{s}) = 1$, we can bound the random variable X_N in (B.21) on both sides as

$$2\rho_{\min} + \frac{\beta}{N}\max_{\boldsymbol{s}\in\mathcal{P}^N} x_{\boldsymbol{s}} \leq X_N \leq \frac{\beta}{N}\max_{\boldsymbol{s}\in\mathcal{P}^N} x_{\boldsymbol{s}}. \tag{B.23}$$

Now take any $\alpha \geq 1$. We can split the expectation in (B.22) into two parts and apply (B.23):

$$\begin{aligned}
\mathbb{E}\left(\mathbb{1}_{|X_N|\geq\alpha}|X_N|\right) &= \mathbb{E}\left(\mathbb{1}_{X_N\geq\alpha}X_N\right) + \mathbb{E}\left(\mathbb{1}_{X_N\leq-\alpha}(-X_N)\right) \\
&\leq \mathbb{E}\left(\mathbb{1}_{\frac{\beta}{N}\max_{\boldsymbol{s}} x_{\boldsymbol{s}}\geq\alpha}[\frac{\beta}{N}\max_{\boldsymbol{s}} x_{\boldsymbol{s}}]\right) \\
&\qquad + \mathbb{E}\left(\mathbb{1}_{2\rho_{\min}+\frac{\beta}{N}\max_{\boldsymbol{s}} x_{\boldsymbol{s}}\leq-\alpha}[-2\rho_{\min} - \frac{\beta}{N}\max_{\boldsymbol{s}} x_{\boldsymbol{s}}]\right) \\
&\leq \sum_{K=1}^{\infty} \alpha(K+1)P(\max_{\boldsymbol{s}} x_{\boldsymbol{s}} \geq \alpha KN/\beta) \\
&\qquad + \sum_{K=1}^{\infty} \alpha(K+1)P(\max_{\boldsymbol{s}} x_{\boldsymbol{s}} \leq -\alpha KN/\beta - 2\rho_{\min}N/\beta),
\end{aligned} \tag{B.24}$$

where to reach (B.24) we have simply decomposed the integrals corresponding to the expectations into a sum of integrals from $K\alpha$ to $(K+1)\alpha$ for $K = 1, 2, \ldots$, and bounded each one.

Let us consider the first probability expression in (B.24). Defining $\Phi(\cdot)$ as the standard Gaussian cumulative distribution function, and exploiting the fact that the ensemble $\{x_{\boldsymbol{s}}\}$ is i.i.d., we have

$$\begin{aligned} P(\max_{\boldsymbol{s}} x_{\boldsymbol{s}} \geq \alpha KN/\beta) &= 1 - P\Big(x_{\boldsymbol{s}} \leq \alpha KN/\beta, \text{ for all } \boldsymbol{s} \in \mathcal{P}^N\Big) \\ &= 1 - \Big(1 - \Phi(-\alpha K\sqrt{N}/\beta)\Big)^{\#\mathcal{P}^N} \\ &\leq \#\mathcal{P}^N \exp\Big(\frac{-\alpha^2 K^2 N}{2\beta^2}\Big), \end{aligned} \tag{B.25}$$

where in reaching (B.25) we have used the inequality $(1-x)^K \geq 1 - Kx$ for any positive integer K and any $x < 1$, and applied the standard Gaussian tail bound $\Phi(-t) \leq \exp(-t^2/2)$ for $t > 0$ (see, *e.g.*, [122, p. 445]). Recall that $\#\mathcal{P}^N = \exp(N \log \lambda_0 + o(N))$. Thus, for all sufficiently large N and sufficiently large α, we can bound the first term on the right-hand side of (B.24) as

$$\begin{aligned} \sum_{K=1}^{\infty} \alpha(K+1) P(\max_{\boldsymbol{s}} x_{\boldsymbol{s}} \geq \alpha KN/\beta) &\leq \sum_{K=1}^{\infty} \alpha(K+1) \exp\Big(2N \log \lambda_0 - \frac{\alpha^2 K^2 N}{2\beta^2}\Big) \\ &\leq \alpha \lambda_0^2 \sum_{K=1}^{\infty} (K+1) \exp\Big(-\frac{\alpha^2 K^2}{2\beta^2}\Big), \end{aligned}$$

which converges to zero as $\alpha \to \infty$. Similar bounds allow us to reach the same conclusion for the second term on the right-hand side of (B.24). It then follows that the uniform integrability condition (B.22) holds for the sequence of random variables in (B.21) corresponding to the free energy density.

Bibliography

[1] L. Addario-Berry, N. Broutin, L. Devroye, and G. Lugosi. On combinatorial testing problems. *The Annals of Statistics*, 38(5):3063–3092, 2010.

[2] A. Agaskar and Y. M. Lu. An uncertainty principle for functions defined on graphs. In *Proc. SPIE Conference on Wavelets and Sparsity (XIV)*, San Diego, CA, 2011.

[3] A. Agaskar and Y. M. Lu. Uncertainty principles for signals defined on graphs: Bounds and characterizations. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc. (ICASSP)*, pages 3493–3496, Kyoto, March 2012.

[4] A. Agaskar and Y. M. Lu. ALARM: A logistic auto-regressive model for binary processes on networks. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2013.

[5] A. Agaskar and Y. M. Lu. Detecting random walks hidden in noise: Phase transition on large graphs. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc. (ICASSP)*, 2013.

[6] A. Agaskar and Y. M. Lu. A fast Monte Carlo algorithm for source localization on graphs. In *SPIE Wavelets*, 2013.

[7] A. Agaskar and Y. M. Lu. A spectral graph uncertainty principle. *IEEE Transactions on Information Theory*, 59(7):4338–4356, July 2013.

[8] A. Agaskar and Y. M. Lu. Optimal hypothesis testing with combinatorial structure: detecting random walks on graphs. In *IEEE Asilomar Conference on Signals, Systems, and Computers*, 2014.

[9] A. Agaskar, C. Wang, and Y. M. Lu. Randomized Kaczmarz algorithms: Exact MSE analysis and optimal sampling probabilities. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2014. *Student Paper Award*.

[10] R. Albert and A.L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, 2002.

[11] E. Arias-Castro, E. Candès, H. Helgason, and O. Zeitouni. Searching for a trail of evidence in a maze. *The Annals of Statistics*, 36(4):1726–1757, August 2008.

[12] E. Arias-Castro, D.L. Donoho, and X. Huo. Near-optimal detection of geometric objects by fast multiscale methods. *IEEE Transactions on Information Theory*, 51(7):2402–2425, July 2005.

[13] C. Asavathiratham. *The influence model: A tractable representation for the dynamics of networked Markov chains*. PhD thesis, Massachusetts Institute of Technology, 2000.

[14] A. Barrat, M. Barthélemy, and A. Vespignani. *Dynamical Processes on Complex Networks*. Cambridge University Press, 2008.

[15] A. Barvinok. A remark on the rank of positive semidefinite matrices subject to affine constraints. *Discrete and Computational Geometry*, 25(1):23–31, 2001.

[16] R. J. Baxter. *Exactly solved models in statistical mechanics*. Academic Press, London, 1982.

[17] M. Belkin. *Problems of learning on manifolds*. PhD thesis, University of Chicago, Chicago, Illinois, 2003.

[18] M. Belkin, I. Matveeva, and P. Niyogi. Regularization and semi-supervised learning on large graphs. In *Learning Theory*, volume 3120, pages 624–638. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[19] V.D. Blondel, J.L. Guillaume, J.M. Hendrickx, and R.M. Jungers. Distance distribution in random graphs and application to network exploration. *Physical Review E*, 76(6, 066101), 2007.

[20] A. Bolstad, B.D. Van Veen, and R. Nowak. Causal network inference via group sparse regularization. *IEEE Transactions on Signal Processing*, 59(6):2628 – 2641, June 2011.

[21] E. Bolthausen and A. Bovier, editors. *Spin Glasses*. Springer, Berlin, 2007.

[22] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[23] A. Buades, B. Coll, and J.M. Morel. A non-local algorithm for image denoising. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, San Diego, CA, 2005.

[24] F. Bénézit, A. Dimakis, P. Thiran, and M. Vetterli. Order-optimal consensus through randomized path averaging. *IEEE Trans. Inf. Theory*, 56(10):5150–5167, October 2010.

[25] L. Calvez and P. Vilbé. On the uncertainty principle in discrete signals. *IEEE Trans. Circuits Syst. II*, 39(6):394–395, June 1992.

[26] E. Candès and J. Romberg. Quantitative robust uncertainty principles and optimally sparse decompositions. *Foundations of Computational Mathematics*, 6:227–254, December 2005.

[27] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory*, 52(2):489–509, 2006.

[28] Emmanuel J. Candès, Philip R. Charlton, and Hannes Helgason. Detecting highly oscillatory signals by chirplet path pursuit. *Applied and Computational Harmonic Analysis*, 24(1):14–40, January 2008.

[29] A. Cattivelli, F.and Sayed. Distributed detection over adaptive networks using diffusion adaptation. *IEEE Trans. Signal Process.*, 59(5):1917–1932, May 2011.

[30] Yair Censor, Gabor T. Herman, and Ming Jiang. A note on the behavior of the randomized Kaczmarz algorithm of Strohmer and Vershynin. *Journal of Fourier Analysis and Applications*, 15(4):431–436, August 2009.

[31] Xuemei Chen and Alexander M. Powell. Almost sure convergence of the Kaczmarz algorithm with random measurements. *Journal of Fourier Analysis and Applications*, 18(6):1195—1214, 2012.

[32] F. R. K. Chung. *Spectral graph theory*. American Mathematical Society, Providence, RI, 1997.

[33] R. Coifman and S. Lafon. Diffusion maps. *Appl. Comput. Harmonic Analysis*, 21(1):5–30, 2006.

[34] R. Coifman and M. Maggioni. Diffusion wavelets. *Appl. Comput. Harmonic Analysis*, 21(1):53–94, July 2006.

[35] A. Comtet, C. Texier, and Y. Tourigny. Products of random matrices and generalised quantum point scatterers. *Journal of Statistical Physics*, 140(3):427–466, 2010.

[36] T. M Cover and Joy Thomas. *Elements of information theory*. Wiley-Interscience, Hoboken, N.J., 2006.

[37] Andrea Crisanti, Giovanni Paladin, and Angelo Vulpiani. *Products of Random Matrices in Statistical Physics*, volume 104. Springer, Berlin, Heidelberg, 1993.

[38] Liang Dai, Mojtaba Soltanalian, and Kristiaan Pelckmans. On the randomized Kaczmarz algorithm. *IEEE Signal Process. Lett.*, 21(3):330–333, Mar. 2014.

[39] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, Philadelphia, PA, 1992.

[40] N. G. de Bruijn. *Asymptotic Methods in Analysis*. Dover Publications, New York, October 2010.

[41] Amir Dembo and Ofer Zeitouni. *Large Deviations Techniques and Applications*. Springer Science & Business Media, November 2009.

[42] B. Derrida. Random-energy model: An exactly solvable model of disordered systems. *Physical Review B*, 24(5):2613, 1981.

[43] A. Dimakis, S. Kar, J. Moura, M. Rabbat, and A. Scaglione. Gossip algorithms for distributed signal processing. *Proc. IEEE*, 98(11):1847–1864, November 2010.

[44] D. L. Donoho and P. B. Stark. Uncertainty principles and signal recovery. *SIAM J. Math. Anal.*, 49(3):906–931, 1989.

[45] David Donoho and Jiashun Jin. Higher criticism for detecting sparse heterogeneous mixtures. *The Annals of Statistics*, 32(3):962–994, June 2004.

[46] D.L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Trans. Inf. Theory*, 47(7):2845–2862, November 2001.

[47] T. C. Dorlas and J. R. Wedagedera. Large deviations and the random energy model. *International Journal of Modern Physics B*, 15(1):1–15, January 2001.

[48] Richard Ellis. *Entropy, Large Deviations, and Statistical Mechanics*. Springer, June 2005.

[49] W. Erb. Uncertainty principles on compact Riemannian manifolds. *Appl. Comput. Harmonic Analysis*, 29(2):182–197, Sep. 2010.

[50] P. Erdős and A. Rényi. On random graphs, I. *Publicationes Mathematicae (Debrecen)*, 6:290–297, 1959.

[51] P. Erdős and A. Rényi. *On the evolution of random graphs*. Akad. Kiadó, 1960.

[52] Vincenzo Fioriti and Marta Chinnici. Predicting the sources of an outbreak with a spectral technique. *arXiv preprint arXiv:1211.2333*, 2012.

[53] G. Folland and A. Sitaram. The uncertainty principle: A mathematical survey. *J. Fourier Analys. Appl.*, 3(3):207–238, May 1997.

[54] H. Furstenberg and H. Kesten. Products of random matrices. *The Annals of Mathematical Statistics*, 31(2):457–469, June 1960.

[55] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[56] C. Godsil and G. Royle. *Algebraic Graph Theory*. Springer, 2001.

[57] S. Goh and T. Goodman. Uncertainty principles and asymptotic behavior. *Appl. Comput. Harmonic Analysis*, 16(1):19–43, Jan. 2004.

[58] Richard Gordon, Robert Bender, and Gabor T. Herman. Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography. *Journal of theoretical Biology*, 29(3):471–481, 1970.

[59] Mark Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, 83(6):1420–1443, May 1978.

[60] Michael Grant and Stephen Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

[61] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, March 2014.

[62] D.K. Hammond, P. Vandergheynst, and R. Gribonval. Wavelets on graphs via spectral graph theory. *Appl. Comput. Harmonic Analysis*, 30(2):129–150, April 2010.

[63] Per Christian Hansen and Maria Saxild-Hansen. AIR tools: A MATLAB package of algebraic iterative reconstruction methods. *Journal of Computational and Applied Mathematics*, 236(8):2167–2178, February 2012.

[64] Gabor T. Herman and Lorraine B. Meyer. Algebraic reconstruction techniques can be made computationally efficient [positron emission tomography application]. *Medical Imaging, IEEE Transactions on*, 12(3):600–609, 1993.

[65] Richard A. Holley and Thomas M. Liggett. Ergodic theorems for weakly interacting infinite systems and the voter model. *The Annals of Probability*, 3(4):643–663, August 1975.

[66] Roger A. Horn and Charles R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, New York, June 1994.

[67] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, October 2012.

[68] Yu I. Ingster and I. A. Suslina. Detection of a signal of known shape in a multichannel system. *Journal of Mathematical Sciences*, 127(1):1723–1736, May 2005.

[69] R. Ishii and K. Furukawa. The uncertainty principle in discrete signals. *IEEE Trans. Circuits Syst.*, 33(10):1032–1034, October 1986.

[70] Philippe Jacquet, Gadiel Seroussi, and Wojciech Szpankowski. On the entropy of a hidden Markov process. *Theoretical Computer Science*, 395(2–3):203–219, May 2008.

[71] S. Kaczmarz. Angenäherte auflösung von systemen linearer gleichungen. *Bull. Internat. Acad. Polon. Sci. Lettres A*, pages 335–357, 1937.

[72] T. Kailath. The innovations approach to detection and estimation theory. *Proc. IEEE*, 58(5):680–695, May 1970.

[73] Tosio Kato. *Perturbation Theory for Linear Operators*. Springer, New York, October 2013.

[74] D. Kazakos. The Bhattacharyya distance and detection between Markov chains. *IEEE Trans. Inf. Theory*, 24(6):747–754, November 1978.

[75] W. O. Kermack and A. G. McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A*, 115(772):700–721, August 1927.

[76] U. A. Khan, S. Kar, and J. Moura. Higher dimensional consensus: Learning in large-scale networks. *IEEE Trans. Signal Process.*, 58(5):2836–2849, May 2010.

[77] D. Klein and M. Randić. Resistance distance. *J. Math. Chem.*, 12(1):81–95, 1993.

[78] Kwangmoo Koh, Seung-Jean Kim, and Stephen Boyd. An interior-point method for large-scale l1-regularized logistic regression. *Journal of Machine learning research*, 8(8):1519–1555, 2007.

[79] Eric D. Kolaczyk. *Statistical Analysis of Network Data: Methods and Models*. Springer, 1 edition, Mar. 2009.

[80] P. Lancaster and M. Tismenetsky. *The Theory of Matrices: with Applications*. Academic Press, New York, 2 edition, 1985.

[81] A.S. Leong, S. Dey, and J.S. Evans. Error exponents for Neyman-Pearson detection of Markov chains in noise. *IEEE Trans. Signal Process.*, 55(10):5097–5103, October 2007.

[82] Bernard C. Levy. *Principles of Signal Detection and Parameter Estimation*. Springer, July 2008.

[83] Andrey Y. Lokhov, Marc Mézard, Hiroki Ohta, and Lenka Zdeborová. Inferring the origin of an epidemy with dynamic message-passing algorithm. *arXiv:1303.5315*, March 2013.

[84] Wuqiong Luo and Wee Peng Tay. Identifying multiple infection sources in a network. In *2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 1483–1489, 2012.

[85] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, 2nd edition edition, 1999.

[86] Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008.

[87] Neri Merhav. Statistical physics and information theory. *Foundations and Trends in Communications and Information Theory*, 6(1-2):1–212, 2010.

[88] M Mézard, G Parisi, and M Virasoro. *Spin Glass Theory and Beyond: An Introduction to the Replica Method and its Applications*. World Scientific, November 1986.

[89] Marc Mézard and Andrea Montanari. *Information, Physics, and Computation*. Oxford University Press, New York, March 2009.

[90] O. Missaoui, H. Frigui, and P. Gader. Land-mine detection with ground-penetrating radar using multistream discrete hidden Markov models. *IEEE Transactions on Geoscience and Remote Sensing*, 49(6):2080 –2099, June 2011.

[91] K. Nakagawa and F. Kanaya. On the converse theorem in statistical hypothesis testing for Markov chains. *IEEE Trans. Inf. Theory*, 39(2):629–633, 1993.

[92] S.K. Narang and A. Ortega. Perfect reconstruction two-channel wavelet filter banks for graph structured data. *IEEE Transactions on Signal Processing*, 60(6):2786–2799, June 2012.

[93] Deanna Needell. Randomized Kaczmarz solver for noisy linear systems. *BIT Numerical Mathematics*, 50(2):395—403, 2010.

[94] Deanna Needell. Randomized Kaczmarz solver for noisy linear systems. *BIT Numerical Mathematics*, 50(2):395–403, 2010.

[95] Deanna Needell and Joel A. Tropp. Paved with good intentions: Analysis of a randomized block Kaczmarz method. *Linear Algebra and its Applications*, 441:199—221, 2014.

[96] Mark Newman. *Networks: An Introduction*. Oxford University Press, USA, 1 edition, May 2010.

[97] Hidetoshi Nishimori. *Statistical Physics of Spin Glasses and Information Processing: An Introduction*. Oxford University Press, New York, September 2001.

[98] Enzo Olivieri and Pierre Picco. On the existence of thermodynamics for the random energy model. *Communications in Mathematical Physics*, 96(1):125–144, 1984.

[99] Dmitry Panchenko. *The Sherrington-Kirkpatrick Model*. Springer New York, New York, NY, 2013.

[100] Bastien Pasdeloup, Réda Alami, Vincent Gripon, and Michael Rabbat. Toward an uncertainty principle for weighted graphs. *arXiv:1503.03291 [cs]*, March 2015. arXiv: 1503.03291.

[101] Mathew Penrose. *Random Geometric Graphs*. Oxford University Press, 2003.

[102] I. Pesenson. Sampling in Paley-Wiener spaces on combinatorial graphs. *Trans. Amer. Math. Soc.*, 360(10):5603–5627, 2008.

[103] Pedro C. Pinto, Patrick Thiran, and Martin Vetterli. Locating the source of diffusion in large-scale networks. *Physical Review Letters*, 109(6):068702, August 2012.

[104] B.A. Prakash, J. Vreeken, and C. Faloutsos. Spotting culprits in epidemics: How many and which ones? In *2012 IEEE 12th International Conference on Data Mining (ICDM)*, pages 11–20, 2012.

[105] Michael Rabbat. On the interplay between topology and signals supported on graphs. In *SPIE Wavelets*. International Society for Optics and Photonics, 2013.

[106] L. Rabiner and B.-H. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1):4–16, 1986.

[107] Pradeep Ravikumar, Martin J. Wainwright, and John D. Lafferty. High-dimensional Ising model selection using l1-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010.

[108] Benjamin Recht and Christopher Ré. Toward a noncommutative arithmetic-geometric mean inequality: Conjectures, case-studies, and consequences. In *Conference on Learning Theory*, 2012.

[109] L. E. Reichl. *A Modern Course in Statistical Physics*. Wiley, August 2009.

[110] R. Tyrrell Rockafellar and Roger J. B. Wets. *Variational Analysis*, volume 317 of *Grundlehren der mathematischen Wissenschaften*. Springer, New York, 1998.

[111] G. Rote. The convergence rate of the sandwich algorithm for approximating convex functions. *Computing*, 48(3-4):337–361, September 1992.

[112] Paul Runkle, Lam H. Nguyen, James H. McClellan, and Lawrence Carin. Multi-aspect target detection for SAR imagery using hidden Markov models. *IEEE Trans. on Geoscience and Remote Sensing*, 39(1):46–55, 2001.

[113] A. Sandryhaila and J.M.F. Moura. Discrete Signal Processing on Graphs. *IEEE Transactions on Signal Processing*, 61(7):1644–1656, April 2013.

[114] L. Scharf and L. Nolte. Likelihood ratios for sequential hypothesis testing on Markov sequences. *IEEE Trans. Inf. Theory*, 23(1):101–109, January 1977.

[115] Eunsoo Seo, Prasant Mohapatra, and Tarek Abdelzaher. Identifying rumors and their sources in social networks. In *Proc. SPIE 8389, Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR III*, May 2012.

[116] D. Shah and T. Zaman. Rumors in a network: Who's the culprit? *IEEE Transactions on Information Theory*, 57(8):5163–5181, 2011.

[117] D.I. Shuman, S.K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, May 2013.

[118] Gilbert Strang and George J. Fix. *An Analysis of the Finite Element Method*. Wellesley-Cambridge Press, 2 edition, 2008.

[119] Robert S. Strichartz. *The Way of Analysis*. Jones & Bartlett Learning, 2000.

[120] Thomas Strohmer and Roman Vershynin. A randomized Kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.

[121] Y. Sung, L. Tong, and H.V. Poor. Neyman-Pearson detection of Gauss-Markov signals in noise: Closed-form error exponent and properties. *IEEE Trans. Inf. Theory*, 52(4):1354–1365, April 2006.

[122] Michel Talagrand. *Mean Field Models for Spin Glasses: Volume I: Basic Examples*. Springer, New York, November 2010.

[123] M. Ting and A.O. Hero. Detection of a random walk signal in the regime of low signal to noise ratio and long observation time. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc. (ICASSP)*, May 2006.

[124] M. Ting, A.O. Hero, D. Rugar, Chun-Yu Yip, and J.A. Fessler. Near-optimal signal detection for finite-state Markov signals with application to magnetic resonance force microscopy. *IEEE Trans. Signal Process.*, 54(6):2049–2062, June 2006.

[125] H.J. Trussell and M.R. Civanlar. Signal deconvolution by projection onto convex sets. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, volume 9, pages 496–499, March 1984.

[126] J. N. Tsitsiklis and V. D. Blondel. The Lyapunov exponent and joint spectral radius of pairs of matrices are hard—when not impossible—to compute and to approximate. *Mathematics of Control, Signals, and Systems (MCSS)*, 10(1):31–40, 1997.

[127] J. Urschel, X. Hu, J. Xu, and L. T. Zikatanov. A cascadic multigrid algorithm for computing the Fiedler vector of graph Laplacians. *J. Comp. Math.*, 33(2):209–226, 2015.

[128] M. Vetterli and J. Kovačević. *Wavelets and Subband Coding*. Prentice Hall, Englewood Cliffs, NJ, 1995.

[129] Erik Volz. SIR dynamics in random networks with heterogeneous connectivity. *Journal of Mathematical Biology*, 56(3):293–310, March 2008.

[130] C. Wang, A. Agaskar, and Y. M. Lu. Randomized Kaczmarz algorithm for inconsistent linear systems: An exact MSE analysis. In *Sampling Theory and Applications (SampTA)*, 2015.

[131] D.J. Watts and S.H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

[132] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

[133] Kai Zhu and Lei Ying. Information source detection in the SIR model: A sample path based approach. *arXiv:1206.5421*, June 2012.

[134] Anastasios Zouzias and Nikolaos M. Freris. Randomized extended Kaczmarz for solving least squares. *SIAM Journal on Matrix Analysis and Applications*, 34(2):773–793, 2013.

[135] Anastasios Zouzias and Nikolaos M. Freris. Randomized Extended Kaczmarz for Solving Least Squares. *SIAM Journal on Matrix Analysis and Applications*, 34(2):773–793, 2013.